中华学习机

紫金Ⅱ系列

APPLEⅡ

汉字软件

朱国江 编著

气象出版社

中华学习机汉字软件

朱国江　编著

气 象 出 版 社

内 容 简 介

本书以国家教育委员会指定推广的优秀机型——中华学习机为选型机(该机与APPLE Ⅱ，紫金 Ⅱ机完全兼容)，系统介绍中华学习机在汉字处理方面的应用，内容广泛，取材新颖，应用性强，适用面广。

本书与《中华学习机编程技巧》(气象出版社，1988年2月出版）和《中华学习机数据处理》(气象出版社，1989年12月出版）等成系列书。

本书适用于从事微机应用、管理人员和广大青少年及微机爱好者自学，也可作为各种类型培训班的教学参考书。

中华学习机汉字软件
朱国江 编著
责任编辑 黄丽荣

中华学习机汉字软件
朱国江 编著
责任编辑 黄丽荣

气象出版社 出版
(北京西郊白石桥路46号)
气象出版社发行 全国各地新华书店经售

*　*　*

开本 787×1092 1/32 印张：14.625 字数：288千字
1990年2月第一版 1990年2月第一次印刷
印数：1—10,000 定价：7.40元
ISBN 7-5029-0329-1 / TP・0017

前　言

国家科委、国家教委、中国科协和电子工业部等单位联合组织开发的中华学习机，已由国营734厂、深圳华明计算机有限公司等10多个厂家生产并开始投放市场，这对我国正在大力普及和推广计算机技术，无疑是一件大好事。

中华学习机小巧玲珑，物美价廉，与APPLE Ⅱ紫金 Ⅱ机兼容，其功能与APPLE Ⅱ相当，并有所增强。它的最大特点是有较强的汉字处理能力，正在日益受到全社会的关注。它特别适用广大中小学教育，将会进入千家万户，只要配上家用彩色或黑白电视机及盒式录音机，广大青少年就可以在家里用上计算机。

为适应广大青少年学习、应用计算机的需要，使广大初、中级计算机管理人员得到切实的帮助，我们撰写了这本小册子，奉献给广大读者以及微机爱好者。

本书以大量的实例，多种选题，深入全面地介绍了中华学习机在汉字处理方面的应用。本书文字通俗，讲解详细，深入浅出，循序渐进。全书起点较低，用于指导自学，基本上可以做到无师自通，而从应用的角度上来说，终点较高，对于有一定基础的读者，是扩大知识、深入学习的有用参考。全书立足于应用、实用，既有一定的原理分析，又有实践经验的总结，并以程序剖析贯穿全书始终，希望能起到开阔视野、启迪思维、抛砖引玉的作用。

本书资料主要来自作者多年编程和教学实践，同时还吸收了国内外的先进经验。书中阐述的方法、原理、技巧、思

路对其它程序设计语言都有参考价值，而实用程序和工具软件可以直接付诸使用或稍作修改后引用。书中全部程序均经中华学习机CEC-I调试通过。对广大APPLE Ⅱ用户，紫金 Ⅱ用户，全书程序均能使用，而其中文汉字部分，用相应的硬汉卡或软汉字输入即可。

在本书的编写过程中，南京大学教授邹进上给予热情支持和鼓励；南京锗厂工程师孙建隆同志，南京大学图书馆馆员宋桂兰同志、为本书汉字编制、程序输入、软件调试付出了大量的劳动，在此一并表示谢意。

编写这种综合性的、面向实用技巧又有大量汉字输入的书籍，还是一次尝试，需要在实践中不断加以完善。鉴于作者水平有限，缺点和错误在所难免，恳切希望同行专家和广大读者不吝赐教。

编　者

1988年 4月25日于南大

目　　录

一、趣 味 程 序

1. 猜迷语

这是给小朋友编制的一个游戏程序——猜迷语。

程序清单如H 1所示:

```
]LIST

5  REM H1
10  PRINT "  猜  迷  语"
20  PRINT "有时落在山腰,有时挂在树梢,"
40  PRINT "有时象个圆盘,有时象把镰刀."
60  PRINT "云儿见它让路,小树见它招手,"
80  PRINT "禾苗见它弯腰,花儿见它点头."
100  PRINT "------------------------"
110  PRINT "小朋友!请你想一想?"
120  INPUT "小朋友回答";A$
140 B$ = "月亮"
150  IF A$ = B$ THEN  PRINT "猜对了": GOTO 240
170  PRINT "你回答错了,是否再猜一次?"
180  INPUT "按Y键再猜,按N键不猜";C$
200  IF C$ = "Y" THEN 120
210  IF C$ = "N" THEN  END
240  END
```

说明: 20句到80句是迷语。120句要求小朋友从键盘输

入答案，用中文回答放在字符串变量A＄中，140句是迷语的正确答案，它放在B＄中，150句是判断，如果小朋友回答正确，计算机显示“猜对了”并转向240句结束；如果小朋友猜的不对，计算机显示“你回答错了，是否再猜一次?”，小朋友继续猜按Y键并回车，经200句判断转向120句，叫小朋友再回答，如果回答对了，结束。如果不猜，则按N键并回车停机。

下面是程序运行的一个实例.

```
]RUN
  猜  迷  语
有时落在山腰,有时挂在树梢,
有时象个圆盘,有时象把镰刀.
云儿见它让路,小树见它招手,
禾苗见它弯腰,花儿见它点头.
------------------------
小朋友!请你想一想?
小朋友回答 HGF
你回答错了,是否再猜一次?
按Y键再猜,按N键不猜Y
小朋友回答 月亮
猜对了
```

注意：本程序运行后，必须在“中文”状态下回答，按自定义键F2处于拼音状态，输入汉语拼音YUE和LIANG才是正确答案，按Y或N键则应用自定义键F 1，使系统处于字母状态。

2. 念数字

下面这个程序，是一个“念数字”的游戏程序。你可以随意输入一个整数（在 0 到99之间），计算机就用汉语拼音印出这个数的念法来。如果输入的数不符合本题要求，计算机会给出“错了”的提示，请求重新输入。

分析：

（1）由于本题要求从键盘输入的数，必须在 0 到99之间的任一个整数，因此，程序中应有防止输入出错的条件判断语句。

不符合规定的输入数是：非整数，小于 0 和大于99的数。

这可以用下述语句处理：

```
70 INPUT X
80 IF INT(X)< > X THEN PRINT“错了”: GOTO 70
82 IF X<0 THEN PRINT“错了”: GOTO 70
84 IF X>99 THEN PRINT“错了”: GOTO 70
```

其中INT(X)< > X，表明输入数与其取整数值不等。

上述语句固然正确，但显得过于烦杂，事实上用逻辑判断语句更为简洁，如：

```
70 INPUT X
80 IF INT(X)< > X OR X<0 OR X>99 THEN PRINT“错了”: GOTO 70
```

这样，80句一句就代替了三个条件语句的判断。

（2）关于本程序的输出，要考虑三种情况：

· 输出是一位数
· 输出是整十位数
· 输出是两位数

我们可以这样处理:

定义A是输入数的十位数字的位，即 A＝INT(X/10)；定义B是输入数的个位数字的位，即 B＝X－10＊A。其中X为输入的 0 —99之间的整数。

故输入数为一位数时，A＝0，打印语句为 A＄(B)；

输入数为整十数时，即 A≠0，B＝0，打印语句为 A＄(A)；“␣”；A＄(10)；

输入数为两位数时,即 A≠0 且 B≠0，打印语句为 A＄(A)；“␣”；A＄(10)；“␣”；A＄(B)。

其中，A＄(I)中记0，1，2，……，9，10的汉语拼音，I＝0,1,2,……，9,10。

(3) 汉语拼音的存贮，可以安排一重循环，用I做循环控制变量，初值为0，终值为10，用字符串数组A＄(I)读取，程序如:

```
30 FOR  I = 0  TO 10
40 READ A$(I)
50 NEXT I
160 DATA“放0,1,2,……，10对应的汉语拼音”
```

即 0：LING　1：YI　2：ER
3：SAN　4：SI　5：WU
6：LIU　7：QI　8：BA
9：JIU　10：SHI

(4) 整个程序采用中文注释，这样使程序的易读性好，

REM语句的注释说明也表明了程序的流程。

完整的程序清单见H 2：

```
]LIST

2  REM H2
5  REM 念数字
10  DIM A$(10)
20  REM 读0,1,2,....,9,10的汉语拚音
30  FOR I = 0 TO 10
40  READ A$(I)
50  NEXT I
60  REM 防止输入出错
70  INPUT "输入0-99间的整数";X
75  IF X = 100 THEN  END
80  IF  INT (X) < > X OR X < 0 OR X > 99 THEN
PRINT "错了": GOTO 70
90  REM  A:输入数的十位数字
95  REM  B:输入数的个位数字
100 A =  INT (X / 10)
110 B = X - 10 * A
120  IF A = 0 THEN  PRINT X;":"A$(B): GOTO 150:
REM    A=0是一位数字
130  IF B = 0 THEN  PRINT X;":";A$(A);" ";A$(10)
: GOTO 150: REM     B=0是整十数
140  PRINT X;":";A$(A);" ";A$(10);" ";A$(B)
145  RESTORE
150  GOTO 70
160  DATA LING,YI,ER,SAN,SI,WU,LIU,QI,BA,JIU,SHI
```

```
]RUN
输入0-99间的整数 34
34:SAN SHI SI
输入0-99间的整数 75
75:QI SHI WU
输入0-99间的整数 10
10:YI SHI
输入0-99间的整数 8
8:BA
输入0-99间的整数 99
99:JIU SHI JIU
输入0-99间的整数 3
3:SAN
输入0-99间的整数 123
错了
输入0-99间的整数 100
```

几点说明：

·A＄(0)也是一个存贮单元，所以DIM A＄(10)，实际上是通知计算机开了11个存贮单元。

·75句是本程序终止标志，输入100时，程序运行结束。

·经过120句和130句处理后，140句实际上是A≠0且B≠0的情况，故不需要再作条件判断，而直接打印输出，此时打印的是对应两位整数的汉语拼音。

3．猜数游戏

这是一个数学游戏程序，计算机出一个数让小朋友猜，

如果小朋友猜的数比计算机出的数大，计算机会提示小朋友“大了,重猜”;如果小朋友猜的数比计算机出的数小，计算机也会提示小朋友“小了，重猜”,直到猜对为止，计算机告诉小朋友“您猜对了!”，并且指出小朋友一共猜了多少次。本程序能提高娃娃们的判断能力。

程序一开始利用随机数发生器产生1—200之间的一个随机整数，这个数放在简单变量C中，但没有显示出来，计算机用中文提示告诉小朋友，有一个数已经产生，其值在1到200之间，请小朋友猜，小朋友从键盘输入一个数S，计算机让S和C比较，如果S＞C，说明小朋友猜的数S比计算机出的数C大,从而计算机作出“大了,重猜”的判断，并返回25句，叫小朋友重猜；反之S＜C，计算机经过比较判断后指出“小了，重猜”的提示，也返回25句让小朋友继续猜，直到猜对为止。为了记录小朋友共猜了多少次，程序中安排了一个计数器B，即B＝B＋1。应该注意的是在小朋友一次还没有猜时，计数器B＝0，它应安排在25句之前。B＝0,是计数器的初始条件，而当小朋友猜了一次后计数器应该加1。

B＝0这个初始化条件是否一定需要呢？对于本程序，可以不要,但只限于中华学习机,或者是兼容机APPLE-Ⅱ以及紫金-Ⅱ等。因为这类机器在重新运行程序时，所有变量均置0。但是有的微机，例如PC-1500机,同样这个程序B＝0一定要有，否则再次运行程序将发生计数错误。

为了使本程序不失一般性，也可以不用B＝0的赋值语句，而改为15 CLEAR，这样对上述各种机型均可运行而不发生错误。用CLEAR语句还有一个优点，它可以代替

对多个变量的清零，从而使程序更为简洁。

程序H 3 的清单和一个运行实例：

```
]LIST

5  REM H3
10 C = 1 +  INT (200 *  RND (9))
15 B = 0
20  PRINT "这是一个猜数游戏，请猜猜看."
22  PRINT "现在已产生了一个1到200之间的随机数."
25  INPUT "S=";S
30 B = B + 1
35  IF S > C THEN  PRINT "大了，重猜": GOTO 25
40  IF S < C THEN  PRINT "小了，重猜": GOTO 25
45  PRINT "您猜对了！您一共猜了";B;"次"
50  END
```

```
]RUN
这是一个猜数游戏，请猜猜看.
现在已产生了一个1到200之间的随机数.
S=123
大了，重猜
S=56
大了，重猜
S=32
小了，重猜
S=44
小了，重猜
S=51
```

```
小了，重猜
S=54
小了，重猜
S=55
您猜对了！您一共猜了7次
```

4. 拆字游戏

输入一个五位整数，要求将该五位数一个一个拆开来，程序如何编制。

例如．输入的五位数是5 3 4 1 2，要求将这五位数拆开来，并成一列打印输出。

方法①

从键盘输入53412，把它赋给变量C，将该数除以10^4，变成5.3412，再取整变成5，将5打印出来，这样就拆开了第一位数，将该数放在数组A(0)中。为了分离第二位数，可采用下面赋值语句：

C＝C－A(0)＊10∧4

计算结果如下：

$$C = 53412 - 5 \times 10^4$$

$$= 53412 - 50000$$

$$= 3412$$

这样五位数就变成4位数。

同上方法，将四位数3412除以10^3．变成3.412，再取整变成4，这样打印出来的就是拆开来的第二位数，再利用C＝C－A(1)＊10∧3，结果变成412。

如此办理，即可一位一位地将原数拆开（分离开）。

根据以上分析，可知乘方的指数是逐次减小的，而数组A的下标是逐渐增加的，它们的间隔（步长）都是1，容易想到用循环语句即可处理上述计算过程，见程序H 4：

```
]LIST

10  REM  H4
20  DIM A(4)
30  INPUT C
40 C = C + 0.01
50  FOR I = 0 TO 4
60 I1 = 4 - I
70 A(I) =  INT (C / 10 ^ I1)
80  PRINT A(I)
90 C = C - A(I) * 10 ^ I1
100  NEXT I
110  END
```

方法②

用字符串函数处理，将数值变成字符，可用INPUT A＄来解决，然后用测字符串函数LEN（A＄），其值就是字符串A＄中的字符个数，将该值赋给变量N，再利用MID＄函数，它可以从一个字符串中取中间若干个字符，例如MID＄（A＄，I，1），就是从A＄中的第I个字符位置起取A＄中的一个字符，改变I的值，就可以随意任取A＄中的字符。显然，上述拆字过程必须由计算机自动完成，而循环就是最好的方法，见程序H 5：

```
]LIST

5  REM H5
10  INPUT A$
20 N =  LEN (A$)
30  DIM X$(N)
40  FOR I = 1 TO N
50 X$(I) =  MID$ (A$,I,1)
60  PRINT X$(I)
70  NEXT I
80  END
```

应该指出的是，程序H5更具有一般性,因为它不仅可以处理数值拆字，而且也可以不必改动任何语句，自由处理字符串中字符的拆字。而程序H4,仅能处理数值拆字，对于字符串却无能为力。

读者不难比较，程序H5比程序H4更为简洁，而且，由于利用了测字符串长度函数LEN(A$)，使程序有了更多的灵活性，只要输入字符串函数A$，LEN(A$)就自动测出字符个数，这样，数组说明语句X$(N)中的N以及循环变量的终值N就都有了确定的值。

对于不只是分离一个字符，MID$函数也具有独到的长处，例如MID$（A$，5,6)就可以从A$中的第5个字符位置起取A$中的6个字符。当然，对于这种分离，也不一定非用MID$函数不可，其它如RIGHT$函数和LEFT$函数也可以用，前者取一个字符串中最右边的若干个字符，后者则取一个字符串中最左边的若干个字符。其

形式分别为RIGHT$(A$, I),LEFT$(A$,J)。

5. 电脑秘书

电脑秘书程序有两个功能:

·平时可以作为一个电子钟。运行程序后，输入当时的时间: 时(H),分(M),秒(S),在屏幕的下部显示带中文的时间，完全象一只电子钟。

·必要时可代替秘书部份工作。例如要在什么时间发布一个什么信息,计算机会准确无误地在该时间发出相应信息,在屏幕上显示中文字幕。

程序编制要点:

(1) 20句，50句以及200—290句组成一个时钟程序，S=S+1，M=M+1，H=H+1分别为计秒、计分、计时，秒满60，分满60,时满24分别由条件语句判断是否进位。当从键盘上输入一个时间后（按时、分、秒次序),屏幕下部即显示时间。

(2) 130句中放了在什么时间发布第几条信息的数据,次序是时、分、秒、代号，如要在8时0分0秒发布第一个信息，则在DATA语句中按8，0，0，1次序存放。而40句为读语句，也按时、分、秒、代号读，变量为H1,M1，S1,X，用80句判断，如果时钟计时到某一时刻，与要发布信息的时间完全相等时，由X值决定发布什么样的信息。信息在屏幕上显示一段时间后再返回计时状态。

(3) 本程序是电脑秘书的原理程序，130句中数据安排是一种示意性说明，实际工作时应按实时情况安排数据，例如上午12点下班，在DATA语句中应为12，0，0，4，这

样，当时钟到12点整时，会在屏幕上显示“到点下班了”的信息。至于第四项X值，亦即按实际情况处理，如有5条信息需要发布，则X＝5。

（4）本程序全部信息是在屏幕上显示，未安排打印输出。另外也可以安排报警语句，使发布信息前，喇叭发声几下；或在发布信息前，先唱一段歌曲，以示醒目，提请大家注意；也可以用程序控制游戏接口开关，自动打开录音机等等。这些工作留给有兴趣的读者自行完成。

程序清单见H 6：

```
]LIST

5  REM H6
10  REM 电脑秘书
15  PRINT  CHR$ (18)
20  INPUT "输入时,分,秒:";H,M,S
40  READ H1,M1,S1,X
50  GOSUB 200
80  IF H = H1 AND M = M1 AND S = S1 THEN 100
90  GOTO 50
100  ON X GOSUB 300,400,500,120
110  GOTO 40
120  VTAB 3: HTAB 2: PRINT "到  点  下  班  了":
 FOR I = 1 TO 1500: NEXT I: END
130  DATA  8,0,0,1,8,0,40,2,8,1,10,3,8,1,40,4
200  FOR K = 1 TO 750: NEXT K
205 S = S + 1
210  IF S = 60 THEN 250
220  IF M = 60 THEN 260
```

```
230  IF H = 24 THEN 270
240  GOTO 280
250 S = 0:M = M + 1: GOTO 220
260 M = 0:H = H + 1: GOTO 230
270 H = 0
280  HOME : PRINT : PRINT : PRINT
285  VTAB 9: HTAB 13: PRINT H;"时";M;"分";S;"秒"
290  RETURN
300  VTAB 3: HTAB 14: PRINT "通  知"
310  VTAB 5: HTAB 2: PRINT "今天上午10点整召开全
公司大会"
315  FOR I = 1 TO 1000: NEXT I
320  RETURN
400  VTAB 3: HTAB 2: PRINT "今天下午2点经理去机
场接外宾"
415  FOR I = 1 TO 1500: NEXT I
420  RETURN
500  VTAB 3: HTAB 2: PRINT "今天下午6点在金陵饭
店举行宴会"
510  FOR I = 1 TO 1500: NEXT I
520  RETURN
```

6. 电脑问路

在大中城市的几个中心闹区，设置几个“电脑问路”亭，对于广大旅客会带来不少方便，他们通过计算机，直接询问城市的交通情况。

下面编制的一个电脑问路程序，是以南京市主要交通路线为例，说明电脑问路的原理。为简单起见，汽车或电车路

线仅安排了起点站和终点站。

程序的运行方法十分简单，敲入01，02，……，31,33，34等任何一个数字代码，屏幕上立即显示该代码对应的行车路线。例如敲入11↙，立即显示：

11路　草场门—樱驼村

它表示乘11路汽车，可以从草场门到樱驼村。

如果将11路途经站名全部填入270句的DATA语句中，则可以了解到整个11路汽车各站情况。

电脑问路程序（见H 7）和运行结果：

```
]LIST

1  REM H7
2  REM  电脑问路
5  REM 南京主要交通路线
20  DIM A$(60)
30  FOR J = 1 TO 13
40  READ A$(J)
45  PRINT A$(J);"  ";
50  NEXT J
52  PRINT
55  INPUT B$
57  IF B$ = "#" THEN 160
58 N = 0
60  FOR J = 1 TO 13
70 N = N + 1
80  FOR I = 1 TO  LEN (A$(J)) -  LEN (B$) + 1
90  IF B$ < > MID$ (A$(J),I, LEN (B$)) THEN 12
0
```

```
100  PRINT B$;" ";"No:";" ";N
110  PRINT "*";A$(J);"*": GOTO 135
120  NEXT I
130  NEXT J
135  RESTORE
140  PRINT "------------------------------------
----"
150  GOTO 55
160  END
200  DATA "01路 南京车站—建康路"
210  DATA "02路 雨花台—岗子村"
220  DATA "03路 江苏路—江苏路"
230  DATA "04路 瑞金路—瑞金路"
240  DATA "05路 汉中门—孝陵卫"
250  DATA "06路 工人医院—中山门"
260  DATA "09路 新街口—中山陵"
270  DATA "11路 草场门—樱驼村"
280  DATA "113路 中央门—水西门"
290  DATA "16路 雨花台—南京西站"
300  DATA "31路 中山码头—建康路"
310  DATA "33路 雨花台—南京西站"
320  DATA "34路 中山码头—新街口"
```

```
]RUN
01路 南京车站—建康路  02路 雨花台—岗子村  03路
 江苏路—江苏路  04路 瑞金路—瑞金路  05路 汉中
门—孝陵卫  06路 工人医院—中山门  09路 新街口—
中山陵  11路 草场门—樱驼村  113路 中央门—水西
门  16路 雨花台—南京西站  31路 中山码头—建康路
  33路 雨花台—南京西站  34路 中山码头—新街口
```

?01
01 No: 1
01路 南京车站一建康路

?02
02 No: 2
02路 雨花台一岗子村

?03
03 No: 3
03路 江苏路一江苏路

?04
04 No: 4
04路 瑞金路一瑞金路

?05
05 No: 5
05路 汉中门一孝陵卫

?06
06 No: 6
06路 工人医院一中山门

?09
09 No: 7
09路 新街口一中山陵

?11
11 No: 8
11路 草场门一樱驼村

```
?113
113 No: 9
*113路 中央门一水西门*
----------------------------------------
?16
16 No: 10
*16路 雨花台一南京西站*
----------------------------------------
?16
16 No: 10
*16路 雨花台一南京西站*
----------------------------------------
?11
11 No: 8
*11路 草场门一樱驼村*
----------------------------------------
?31
31 No: 11
*31路 中山码头一建康路*
----------------------------------------
?33
33 No: 12
*33路 雨花台一南京西站*
----------------------------------------
?34
34 No: 13
*34路 中山码[illegible]一新街口*
----------------------------------------
?#
```

说明：

程序H7编制中，应特别注意DATA语句中的代码，例如，南京市有13路公共汽车，但没有113路，如果安排13路，则本程序查找不出对应的行车路线；又如南京市有1，2，3，4，……9路车，但没有01，02，03，04，……，09路车，但若安排3，2，6，9，等，则不是查错，就是找不到。

原因是本程序在查找中，不仅查找代码，而且查找中文汉字，而有些中文汉字的代码（它们是用机器语言编制的），可能和代码一致，结果按代码查找不出来。

所以用本程序编制电脑问路程序时，一定要反复调试，以确定代码，否则面貌全非，引错了路。

如将上述程序中110句后面的GOTO135删去，则电脑问题更加灵活。例如，问到南京西站如何走，则输入南京西站四个字后，电脑指示既可以坐16路汽车，也可以乘33路电车。

```
]RUN
01路 南京车站一建康路  02路 雨花台一岗子村  03路
 江苏路一江苏路  04路 瑞金路一瑞金路  05路 汉中
门一孝陵卫  06路 工人医院一中山门  09路 新街口一
中山陵  11路 草场门一樱驼村  113路 中央门一水西
门  16路 雨花台一南京西站  31路 中山码头一建康路

  33路 雨花台一南京西站  34路 中山码头一新街口
?中山
中山 No: 6
*06路 工人医院一中山门*
```

```
中山 No: 7
*09路 新街口—中山陵*
中山 No: 11
*31路 中山码头—建康路*
中山 No: 13
*34路 中山码头—新街口*
--------------------------------------
?新
新 No: 7
*09路 新街口—中山陵*
新 No: 13
*34路 中山码头—新街口*
--------------------------------------
?南京西站
南京西站 No: 10
*16路 雨花台—南京西站*
南京西站 No: 12
*33路 雨花台—南京西站*
--------------------------------------
?#
```

7. 电脑侦探

计算机具有逻辑判断和推理的能力，因而也就具备了一定的逻辑思维能力，正因为如此，人们常常称计算机叫“电脑”。

计算机可以看病，预报天气，也可以破案，下面我们通过两个实例，看看电脑的本领。

例1，A、B、C三人在一起，A指责B正在撒谎，B指责C正在撒谎，而C指责A和B都在撒谎。试编制一个BASIC程序，帮助你做出正确的判断，究竟谁在撒谎？

分析：

· 对这个具体问题，因为只有撒谎和不撒谎两种可能，所以可以作一个逻辑上的假定：撒谎叫1，说实话叫0。

· 对每一个人来说，都有撒谎和不撒谎两种可能，因此三个人（三个变量），就有8种不同的情况，即：

A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

· 本题中要求找出撒谎者是谁，即A、B、C中谁是“1”。

根据题意我们来逐一分析：

① A指责B正在撒谎

这有两种可能，若A说的真话，那么B是说谎者；若A

说的假话，则B说的是真话。即：

若A＝0，则B＝1；

若A＝1，则B＝0；

上述情况可以用ABS（A－B）＝1来综合。显然，A＝B＝1（都说谎）和A＝B＝0（都说实话），不符合题意要求，即ABS（A－B）＝0应该排除。

② B指责C正在撒谎

和上面分析一样，“既然”B指责C正在撒谎”，那么若B＝0（说实话），则C＝1（撒谎）；若B＝1，则C＝0，即ABS（B－C）＝1。而ABS（B－C）＝0，不符合题意。

③ C指责A和B正在撒谎

同理：这句话可以写成ABS(C－A＊B)＝1。而ABS(B－C)＝0，应该舍去。

现在，我们可以安排写程序了。

A、B、C都有撒谎（记作1）或说实话（记为0）两种可能，共有2^3＝8种情况，因此可以安排三重循环，取A、B、C作为循环变量，分别对应各人说谎与否的两种情况，即：

```
10 FOR A=0 TO 1
20 FOR B=0 TO 1
30 FOR C=0 TO 1
   :
   :
   :
   :
```

```
100 NEXT C
110 NEXT B
120 NEXT A
```

上面三个条件，要拿来逐一验证8种情况，只要有一条不符，就应排除，即在循环体中安排下述三句：

```
]LIST 40,60

40  IF  ABS (A - B) = 0 THEN 100
50  IF  ABS (B - C) = 0 THEN 100
60  IF  ABS (C - A * B) = 0 THEN 100
```

为什么要安排上述三句呢？

我们可以拿第一个条件来分析：

条件1是若A＝0，则B＝1；若B＝0，则A＝1。

我们用"反证法"，若不满足上述条件，就是ABS(A－B)＝0。

同时写成ABS(A－B)＝0还有两层意思，因若A＝1，B＝1和A＝0，B＝0，这两种情况，都不是题意要求，所以写成ABS(A－B)＝0，就包括了ABS(1－1)＝ABS(0)＝0和ABS(0－0)＝ABS(0)＝0这两种情况。既然不满足，就应排除，所以用IF……THEN……语句，而THEN后面的行号显然应该找到100。

至于50句，60句，同上分析，不再多述。

排除了所有不符合条件（题意）的各种可能，剩下来的就是有可能说谎的人，再进一步判断，即：

```
]LIST 70,90

70  IF A = 1 THEN  PRINT "A"
80  IF B = 1 THEN  PRINT "B"
90  IF C = 1 THEN  PRINT "C"
```

完整的程序见H8：

```
]LIST

5  REM H8
10  FOR A = 0 TO 1
20  FOR B = 0 TO 1
30  FOR C = 0 TO 1
40  IF  ABS (A - B) = 0 THEN 100
50  IF  ABS (B - C) = 0 THEN 100
60  IF  ABS (C - A * B) = 0 THEN 100
70  IF A = 1 THEN  PRINT "A"
80  IF B = 1 THEN  PRINT "B"
90  IF C = 1 THEN  PRINT "C"
100  NEXT C
110  NEXT B
120  NEXT A
```

```
RUN
A
C
```

由此可知，真正说谎的是A和C，而B是老实人。

程序H8显得比较冗长，再改进一下则简短多了，见程序H9：

```
]LIST

5  REM H9
10  FOR A = 0 TO 1: FOR B = 0 TO 1: FOR C = 0 TO
 1
20  IF  ABS (A - B) = 0 OR  ABS (B - C) = 0 OR
ABS (C - A * B) = 0 THEN 60
30  IF A = 1 THEN  PRINT "A"
40  IF B = 1 THEN  PRINT "B"
50  IF C = 1 THEN  PRINT "C"
60  NEXT C: NEXT B: NEXT A
```

读了以上程序，有人不禁要问，这不是人编程序吗？那里是计算机破案呢？

对于这个问题，要弄清人与机器之间的关系。

人不编程序——机器作不出判断；

机器根据人编的程序——才能作出判断。

这里人是第一位的，机器是第二位的。人仅根据题意要求，编写了程序，但未作判断。人要作出判断，必需花费很多时间和精力，要考虑各种可能情况，又要排除不合理的因素；既要联贯思索，又要科学推理。而当实际问题比较复杂时，人的智力不可能一下子涉及到问题的全部细节，特别是倘有多种复杂的因素，多个相互制约的关系时，常常难于做出正确的判断。所以遇到这类情况，最好的办法是求助于计算机帮忙。编一个程序，让计算机去做大量的分析、比较、判断的工作。最后所得结果正确与否，还取绝于人编的程序。

下面一个例题，情况就复杂多了，仅靠人来处理，往往一下子得不到结果。

例2，某侦察队长接到一项紧急任务，要求在最短的时间内侦破一个案子。已知和案件有关的是代号为A、B、C、D、E和F共6人，掌握的线索如下：

(1) A、B两人中至少有一人参与作案；

(2) A、D两人不是同案犯；

(3) A、E、F三人中有二人作案；

(4) B、C两人或者同时参与作案，或者都与本案无关；

(5) C、D中有一人也仅有一人系案犯；

(6) 若D未作案，则E也不是案犯。

分析：

本例案情比较复杂，对每个嫌疑分子，都有作案（定为1）和不作案（定为0）两种可能，因此，就有$2^6=64$种情况，需要分析、判断。显然，这64种情况，要人来逐一考虑是太复杂了，可以由计算机来处理，例如，安排六重循环，初值为0终值为1，分别区分不作案和作案的各种可能。

·对线索(1)，A、B二人至少有一人参与作案，用逻辑术语来说就是：

或者A作案，B未作案，即A＝1，B＝0；

或者B作案，A未作案，即B＝1，A＝0；

或者A、B均作案，即A＝1，B＝1。

上面三种情况，用逻辑表达式表示，即：

A OR B＝1

显然，A OR B＝0，不符合题意要求。

用程序来描述就是：

```
10  FOR  A=0  TO  1
```

```
20 FOR B=0 TO 1
30 IF A OR B=0 THEN 50
40 PRINT A;" ";B
50 NEXT B
60 NEXT A
```

程序中30句，也可以这样写：

```
30 IF A+B=0 THEN 50
```

或者：

```
30 IF SGN(A+B)=0 THEN 50
```

·对线索(2)，A、D二人不可能同时作案，换句话说 IF A AND D=1 成立，是不符合题意的。

这个判断也可以写成：

```
IF A*D=1
```

或者：

```
IF A+D=2
```

·对线索(3)，A、E、F三人中有二人作案，即 IF A+E+F<>2 成立，是不符合题意的。

这个判断也可以写成：

```
IF SGN(A*E+A*F+E*F)=0
```

·对线索(4)，B、C两人或者同时作案，或者均未作案，可以写成：

```
IF (B+C)/2=INT((B+C)/2)
```

而IF(B+C)/2<>INT((B+C)/2)是不符合题意的。

同样，也可以写成：

```
IF ABS(B-C)=1，不满足题意。
```

·对线索(5)，C、D 中有且仅有一人作案，反过来说，就是，IF C + D< >1，不符合题意，也可以写成 IF ABS(C－D)＝0，不满足条件。

·对线索(6)，如果D未作案，则E也未作案，可以写成D＝1，E＝1是不符合条件的。

总之，我们可以用逻辑表达式，或者数学表达式来描述上述各种线索，而用反证的方法逐一排除不满足题意要求的条件，用循环的方法逐一投试，即可迅速解题抓住作案者。

今给出两种破案程序H10和H11，解题思路一样，但条件语句写法不同。

最后结果都是代号为A、B、C、F四人是案犯。

显然，本题如果不编程序，而由人来判断，则是比较困难的。因此，在以后的编程中，我们可以根据题目的要求，在有可能利用计算机协助我们解题时，尽可能巧用计算机，这会使你减少许多计算工作，以及对一些复杂问题的判断、分析。

程序H10：

```
]LIST

2  REM H10
5  REM 电脑侦探2
10  FOR A = 0 TO 1
20  FOR B = 0 TO 1
30  IF A + B = 0 THEN 190
40  FOR C = 0 TO 1
50  IF (B + C) / 2 < > INT ((B + C) / 2) THEN
180
60  FOR D = 0 TO 1
```

```
70  IF A + D = 2 THEN 170
80  IF C + D < > 1 THEN 170
90  FOR E = 0 TO 1
100  IF D = 1 THEN 120
110  IF E = 1 THEN 160
120  FOR F = 0 TO 1
130  IF A + E + F < > 2 THEN 150
140  PRINT "A:";A;" ";"B:";B;" ";"C:";C;" ";"D:"
;D;" ";"E:";E;" ";"F:";F
145  GOTO 210
150  NEXT F
160  NEXT E
170  NEXT D
180  NEXT C
190  NEXT B
200  NEXT A
210  END

]RUN
A:1 B:1 C:1 D:0 E:0 F:1
```

程序H11:

```
]LIST

2  REM H11
5  REM  电脑侦探3
10  FOR A = 0 TO 1
```

```
20  FOR B = 0 TO 1
30  IF  SGN (A + B) = 0 THEN 190
40  FOR C = 0 TO 1
50  IF  ABS (B - C) = 1 THEN 180
60  FOR D = 0 TO 1
70  IF A * D = 1 THEN 170
80  IF  ABS (C - D) = 0 THEN 170
90  FOR E = 0 TO 1
100  IF D = 1 THEN 120
110  IF E = 1 THEN 160
120  FOR F = 0 TO 1
130  IF  SGN (A * E + A * F + E * F) = 0 THEN 15
0
135  PRINT "A";",";"B";",";"C";",";"D";",";"E";"
,";"F"
140  PRINT A;",";B",";C;",";D;",";E;",";F
145  GOTO 210
150  NEXT F
160  NEXT E
170  NEXT D
180  NEXT C
190  NEXT B
200  NEXT A
210  END
```

```
]RUN
A,B,C,D,E,F
1,1,1,0,0,1
```

8. 电脑医生

计算机可以看病，这已经为越来越多的人知道，但是，它为什么会代替医生诊断，却不是多数人所了解的。计算机诊断属于人工智能中的专家系统(EXPERT SYSTEM)领域，而专家系统已有很广泛而又很重要的应用，它几乎遍及人类生活的各个方面。

我们这里介绍的是一个比较简单的供娱乐和示教用的电脑医生诊断程序，它可以根据病人的不同“症状”，诊断出16种疾病，并开出“处方”。所以要举这样一个例子，是因为它简单明了，易为人们所接受，并且是用BASIC语言编写，这样，就会使更多的人了解专家系统的基本工作原理。

“天有不测风云，人有旦夕祸福”，计算机能预报天气，也能给病人看病，开药方，这不是太“神”了吗？这到底是怎么一回事呢？人们都想了解这个问题。

现在，我们先列出电脑医生诊断程序H12，然后逐段逐段地剖析。

```
]LIST

5  REM H12
10  DIM A(4),B(15),T(15,2)
20  PRINT "同志,您好!我是电脑医生,您叫什么名字?"
;
30  INPUT "输入您的名字";N$
40  PRINT N$;"同志,您的体温是多少?";
50  INPUT "输入体温";T1
60  PRINT "请问您有哪些症状(您可选择下列症状前的
```

```
号码,例如1245)?"
70  PRINT "1.头痛    2.咳嗽"
80  PRINT "3.腹痛    4.发冷"
90  PRINT "5.恶心    6.泻肚"
100  PRINT "7.乏食    8.呕吐"
110  PRINT "9.乏力"
120  INPUT "输入症状号码";C
125 C = C + 0.01
130  FOR I = 0 TO 4
140 I1 = 4 - I
150 A(I) =  INT (C / 10 ^ I1)
160 C = C - A(I) * 10 ^ I1
170  NEXT I
180  FOR I = 0 TO 3
190  FOR J = I + 1 TO 4
200  IF A(I) < A(J) THEN 230
220 X = A(I):A(I) = A(J):A(J) = X
230  NEXT J
240  NEXT I
250 C = A(0) * 10000 + A(1) * 1000 + A(2) * 100
+ A(3) * 10 + A(4)
260  RESTORE
270  FOR I = 0 TO 15
280  READ B(I),T(I,1),T(I,2)
290  NEXT I
300  FOR I = 0 TO 15
310  IF  ABS (C - B(I)) > 0.01 THEN 380
320  IF T1 < T(I,1) THEN 380
330  IF T2 > T(I,2) THEN 380
340  FOR J = 0 TO I: READ I$,D$: NEXT J
350  PRINT N$;"同志,您得了";I$
```

```
360  PRINT "     处     方"
370  PRINT D$: END
380  NEXT I
390  PRINT "很抱歉,这病我看不了,请另请医生看"
500  DATA  12,37,38,14,38,43,145,38,43,124,38,43
,1245,38,43,24,37,43,36,36.5,37,5789,36.5,40,358
,36.5,40,189,38,43,369,38,40,1,36.5,38,7,36.5,37
,2,36.5,37,6,36.5,37,0,37,39
510  DATA  感冒,桑菊感冒片(4-6片/次 2次/日)
520  DATA  流感,银翘解毒片(4-6片/次 2次/日)
530  DATA   流感,银翘解毒片(4-6片/次 2次/日)
540  DATA  流感,银翘解毒片(4-6片/次 2次/日)
550  DATA  流感,银翘解毒片(4-6片/次 2次/日)
560  DATA 肺炎,青霉素(肌肉注射)
570  DATA  肠炎,四环素(1片/次 4次/日)
580  DATA   肝炎,注意营养
590  DATA 阑尾炎,住院开刀
600  DATA 乙型脑炎,立即住院治疗
610  DATA  痢疾,痢特灵(1片/次 3次/日)
620  DATA  头痛,去痛片(2片/次 3次/日)
630  DATA  消化不良,食母生(4片/次 3次/日)
640  DATA  咳嗽,止咳糖浆(10mm 3次/日)
650  DATA 腹泻,活性碳片(4片/次)
660  DATA  低烧,解热止痛片(2片/次 3次/日)
```

本程序共分六段编写，概略说明如下：

(1) 10—120 句：用 PRINT 语句提示中文信息，电脑

医生询问病人姓名，体温和症状，病人依次作出相应的回答。

（2）125—170句：计算机将症状组合的号码进行拆字处理，如“2145”表示患者有头痛、咳嗽、发冷、恶心等四种症状，将号码拆开后分别存入A(0)—A(4)中，结果A(0)—A(4)中分别是“0”，“2”，“1”，“4”，“5”。

（3）180—250句：将数组元素A(0)—A(4)中的症状代码按其数值大小进行排序，并重新存贮在A(0)—A(4)，此时它们的值分别是“0”，“1”，“2”，“4”，“5”。经250句处理合并，赋给变量C，此时C中的值为“1245”，它和“2145”表示同样的症状组合。

（4）260—390句：根据病人体温和症状进行病情诊断，然后给出诊断结果（病名），并开出处方。数组B，T中分别存放症状组合和每种症状对应的体温上、下限，只有输入的症状组合C和体温T1与B，T中的元素相等时，才能看病，否则则认为该病无法诊断。

（5）500语句中放的是症状组合以及每种症状组合所对应的体温下限和上限。注意症状组合的代码是由小到大排列的。症状组合，体温下限和体温上限也是按次序顺序存放的。

（6）510—660句：每一个DATA语句放一种病名和对应该病的诊断处方，中间只用一个“，”号隔开，括弧中不能有“，”号。病名和处方顺序存放，病名在前，处方在后。

变量设置：

N$——患者姓名，键盘输入中文或英文。

T1——患者体温，到小数第一位。

C ——症状，最多四位数，可以不按顺序。

I，J——循环控制变量。

A(I)——存放症状组合C拆开的数码。

B(I)——存放症状组合。

T（I，J）——存放体温值，T（I，1）存体温下限，T(T,2)存体温上限。

I $ ——病名。

D $ ——处方。

主要程序段分析：

(1) 拆字

程序见H12-1：

```
]LIST 100,170

100  REM H12-1
110  DIM A(4)
120  INPUT "输入症状号码";C
125 C = C + 0.01
130  FOR I = 0 TO 4
140 I1 = 4 - I
150 A(I) =  INT (C / 10 ^ I1)
160 C = C - A(I) * 10 ^ I1
170  NEXT I
```

运行上述程序后在A(0)—A(4)中,分别存放了“5”，“3”，“4”，“1”，“2”五个字码。

为什么输入的一组代码要一个一个地拆开来呢？主要是为排序作准备。例如53412这五个数码,它们不是按顺序排列

的，为此，首先将它们一个字符一个字符地拆开来，然后才可以用排序的方法将它们按1,2,3,4,5顺序排列。

(2) 排序

对一组数据或一组字符串，按从大到小或从小到大的次序进行排列，称为排序(SORT)，又称分类。它是数据处理中最基本的工作之一。

排序的方法很多，其中流程比较简单，容易为初学者接受的是枚举分类法，它的基本思想是逐一比较。

例如，有N个数，要求从小到大顺序排列。首先把N个数放在数组A(N)中，让A(I)与后面的一个存贮单元内容A(J)相比较；

如果A(I)<A(J)，两者不交换，再让A(J+1)与A(I)比较；

如果A(I)>A(J)，两者交换，方法是设置一个中间变量X，用X=A(I)：A(I)=A(J)：A(J)=X来实现交换，交换后再让A(J+1)与A(I)比较；

注意：A(I)与A(J)比,实际上是A(I)要与(N-1)个数比,这是第一轮的情况。在第一轮比较中A(I)=A(1),比较的结果就是求出N个数中较小的一个数，并把它存在A(1)中。

以后的工作，就是对剩下来的（N-1）个数，进行同样的工作，每次都把剩下来的数中最小的存入A(I)中，对第二轮就是存入A(2)中。一共进行（N-1）轮比较，就可以把N个数按小到大顺序排列。

程序如H12-2所示：

```
]LIST170,240

170  REM H12-2
180  FOR I = 0 TO 3
190  FOR J = I + 1 TO 4
200  IF A(I) < A(J) THEN 230
220 X = A(I):A(I) = A(J):A(J) = X
230  NEXT J
240  NEXT I
```

程序H12-2是针对本题而设置的，一共是5个数，运行约最后结果，是将症状代码按其数值大小放在A(0)—A(4)中。

(3) 合拼

这项工作主要由250句来完成，事实上，A(0)中存放的是万位数的位，A(0)×10000才是万位数，A(1)中存放的是千位数的位，A(1)×1000才是千位数，其余依次类推。

经250句处理后，变量C中的值就是已经排好顺序的病症组合的数码。

例如，120句输入症状号码为1432，经125—170句程序段处理后存贮情况为：A(0)=0，A(1)=1，A(2)=4，A(3)=3，A(4)=2，经120—240句程序段处理后存贮情况为：A(0)=0，A(1)=1，A(2)=2，A(3)=3，A(4)=4，最后经250句处理后，C=1234。

现在，可以解释为什么本程序要经过拆字、排序、合併的处理，目的是为了与500句中症状组合相比较和检索，因为500句中的症状组合号码是按序排列的。

综合所述，拆字为了排序，排序为了合并，合并为了检索，以上三段反映了本程序的主要流程，实际上也就是设计思想。

(4) 诊断和开处方

程序由260—390句组成。

这是整个程序的核心。

为便于分析，现列出260—390程序段，见程序H12-3：

```
]LIST250,380

250  REM H12-3
260  RESTORE
270  FOR I = 0 TO 15
280  READ B(I),T(I,1),T(I,2)
290  NEXT I
300  FOR I = 0 TO 15
310  IF  ABS (C - B(I)) > 0.01 THEN 380
320  IF T1 < T(I,1) THEN 380
330  IF T2 > T(I,2) THEN 380
340  FOR J = 0 TO I: READ I$,D$: NEXT J
350  PRINT N$;"同志,您得了";I$
360  PRINT "     处     方"
370  PRINT D$: END
380  NEXT I
```

270—290句是读存500句中的信息，这样症状组合存在B(I)中，对应该症状组合的体温下限存在T(I, 1)中，而体温上限存在T(I, 2)中，I=0,1,2,……，15共16种病情诊断依据。

300—380句是一个大循环，首先用病人输入的症状组合(它已经按数值大小由小到大排列好)C与B中的每种症状组合进行比较，若C与B中的某元素相等，再用体温T1与T中对应的体温上、下限比较，若T1在此上、下限内，则找出对应的诊断结果（病名，I$）及处方（D$），并显示出来。若C与所有B中的症状组合均不符合，或虽符合但体温不在其对应的上下限内，则认为该病电脑医生无法诊断。

260句RESTORE是恢复数据区语句。

390句是给出无法诊断的提示。

程序运行方法：

程序运行后，“患者”可根据电脑医生的询问输入姓名和体温，然后根据列出的“症状清单”，输入所患症状。可以同时输入几个病状，但不超过4个。电脑医生根据输入症状和体温进行诊断并开出处方。对于患者输入的症状组合如果太奇怪，则电脑医生无法诊断，并给出“很抱歉，这病我看不了，请另请医生看”的提示。症状输入，不一定按次序，例如“189”和“198”表示同样症状组合，会得出同样的诊断结果。但同一个症状只能输一次，例如“114”，电脑医生无法诊断。如果无症状，可输入“0”，但要输入体温。

程序运行实例：

```
]RUN
同志,您好!我是电脑医生,您叫什么名字?输入您的名字
林琳
林琳同志,您的体温是多少?输入体温38.8
请问您有哪些症状(您可选择下列症状前的号码,例如12
45)?
```

```
1.头痛      2.咳嗽
3.腹痛      4.发冷
5.恶心      6.泻肚
7.乏食      8.呕吐
9.乏力
输入症状号码128
很抱歉,这病我看不了,请另请医生看

]RUN
同志,您好!我是电脑医生,您叫什么名字?输入您的名字
SMITH
SMITH同志,您的体温是多少?输入体温41
请问您有哪些症状(您可选择下列症状前的号码,例如12
45)?
1.头痛      2.咳嗽
3.腹痛      4.发冷
5.恶心      6.泻肚
7.乏食      8.呕吐
9.乏力
输入症状号码 189
SMITH同志,您得了乙型脑炎
    处      方
立即住院治疗
]RUN
同志,您好!我是电脑医生,您叫什么名字?输入您的名字
林微微
林微微同志,您的体温是多少?输入体温39.2
请问您有哪些症状(您可选择下列症状前的号码,例如12
45)?
1.头痛      2.咳嗽
3.腹痛      4.发冷
```

5.恶心　　6.泻肚
7.乏食　　8.呕吐
9.乏力
输入症状号码 358
林微微同志,您得了阑尾炎
　　处　　方
住院开刀

]RUN
同志,您好!我是电脑医生,您叫什么名字?输入您的名字
苏铭
苏铭同志,您的体温是多少?输入体温39.1
请问您有哪些症状(您可选择下列症状前的号码,例如12
45)?
1.头痛　　2.咳嗽
3.腹痛　　4.发冷
5.恶心　　6.泻肚
7.乏食　　8.呕吐
9.乏力
输入症状号码 369
苏铭同志,您得了痢疾
　　处　　方
痢特灵(1片/次 3次/日)

9. 破译密码

一个藏有绝密文件的保险柜，自1948年，很长时间无法打开。几经调查，终于了解到这一保险柜的开锁密码每天都不同，其变化规律如下：

① 保险柜的密码是一个六位数；

② 把六位数按前后顺序拆成两个三位数A和B，则它们之差正好等于当天的星期数（例如A－B＝4，表示星期四）；

③ 星期数为偶数时，A、B之和正好等于1948；

④ 星期数为奇数时，A、B之和正好是当年年份的约数。

现在，我们来编制一个 BASIC 程序，要求尽快地破译这个开锁密码。

题意分析：

密码是一个六位数，又按前后顺序分成两组A和B，可知前三位数A的第一位最小应是1，二、三两位最小是0；后三位数B的第一位最大为9，二、三两位最大也是9，这样，就可大致划分六位数的变化范围。容易想到若取I做循环变量，初值为100，终值为999，安排一个循环，即可“扫描”所有的三位数。

密码每天都不同，而且是以星期数来区分的，所以断定密码一共有7组，即星期一是一组密码，星期二是一组密码，……，星期七（日）又是一组密码。若取J作为另一个循环变量，循环次数为1到7，从而表示每天的星期数，即J＝1就是星期一，J＝2就是星期二，……等等。

根据A、B之差就是当天的星期数，即A－B＝J，故

可以取A＝J＋I：B＝I。

由星期数J为奇数时（即在星期一、三、五、七（日）4天中），两数A、B之和正好是当年年份的约数这一条件，可以写成关系式：1948/(A＋B)＝INT(1948/(A＋B))

由星期数J是偶数时（即星期二、四、六），A、B之和为1948，则可以直接写成：A＋B＝1948。

至此，我们已逐一考虑了满足密码变化的四个条件，故容易写出如下程序，程序名为PROGRAM-ONE。

```
]LIST

3  REM    破译密码一
5  REM PROGRAM-ONE
10  DIM A$(7)
20  FOR K = 1 TO 7
30  READ A$(K)
40  NEXT K
50  DATA 星期一,星期二,星期三,星期四,星期五,星期
六,星期日
60  FOR I = 100 TO 999
70  FOR J = 1 TO 7
80 A = J + I:B = I
90  IF (J / 2 < > INT (J / 2)) AND (1948 / (A
+ B) = INT (1948 / (A + B))) THEN PRINT J;"-";
A$(J);" ";A;B: GOTO 110
100  IF A + B = 1948 THEN PRINT J;"-";A$(J);" "
;A;B
110  NEXT J
120  NEXT I
130  END
```

程序中10—50句，是把星期一（MUN），……，星期日（SUN），存入A＄(K)中（K＝1，2，……，7）。

90行是用逻辑表达式写的，它的意思很清楚，如果星期数J不是偶数（即奇数），同时A＋B又是1948的约数的话，则打印出星期数J，相应的中文汉字和A、B两组数码，然后再循环找其余各组。

如果不满足90句条件，则100句就是偶数的情况。

如果90句、100句的条件均不满足，则不符合题意，再继续搜索。

把程序PROGRAM-ONE敲入计算机，让它运行去吧！结果肯定会有，但是你必需要有足够的耐心，它的速度一定是很慢的。

原因何在？循环次数高达899×7＝6293次。

乘计算机忙于“破译密码”的时间，我们不妨仔细地研究一下，看一看程序PROGRAM-ONE还有没有值得推敲和改进的地方。

根据前面分析，密码只有7组，满足所有四个条件的只有也仅仅有7个记录，显然，6293次循环中，大量的是无效循环。

众所周知，循环部份执行速度的快慢，对整个程序运行时间的影响是至关重要的。而减少循环的次数，特别是减少不必要的循环，是提高速度的关键。为此，减少循环嵌套、增大循环步长以及缩短初值与终值的差距等，都是值得一试的方法。

对照程序PROGRAM-ONE，我们可以分析一下那些方法可取，那些方法又是不可行的。

首先，我们认为内循环70—110句是不能省略的，因为循环变量J是星期数，1到7一个不能少，步长也只能是1。这样，增大循环步长，缩小初、终值差值的办法是行不通的。

其次，本程序一共有两重循环，外循环是取遍所有的三位数，内循环是表示每天的星期数，二者缺一不可，因而减少循环层次的办法也无法实现。

最后，我们只有把精力放在外循环的初、终值的取值范围上，看一看有没有办法，找到提高速度的措施。显然，最主要之点，就是设法尽量减少无效循环的次数。

现在，我们不妨把本题的四个条件，综合起来进行分析和思索，并尽可能利用计算机帮助我们作出一些判断。

由题意，我们容易写出以下三个关系式：

$A-B=J$　　（当 $J=1,2,3,4,5,6,7$ 时）　　（1）

$A+B=1948$　　（当 $J=2,4,6$ 时）　　（2）

$1948/(A+B)=K$　（当 $J=1,3,5,7$　K为正整数时）（3）

现在的问题是如何减少60句I从100到999中无效循环的次数，换句话说就是如何判断三位数A、B的变化范围。这个问题初看起来，似乎无法下手，但是仔细分析一下，它还是有规律可寻的。

第一，我们可以由（3）式判断K取正整数时的变化范围（若K不是正整数，则不符合题意），从而确定A、B之和的变化范围（同理，A＋B是正整数时有效，反之不符合题意）。

如何判断K的变化范围呢？

我们可以作一个近似估算，若 $K=10$，则由（3）式知 $A+B=194.8$，这显然是不符合事实的；其一，A＋B不能

有小数；其二，A、B均为三位数，A＋B最小是100＋100＝200，所以K值最大不会超过10。

为了正确估算A＋B的变化范围，我们可以编一个程序来求解，其程序见PROGRAM－TWO。

```
]LIST

195  REM  PROGRAM-TWO
200  FOR K = 1 TO 10
210 X = 1948 / K
215  IF X =  INT (1948 / K) THEN 220
218  GOTO 230
220  PRINT "K=";K" ";"X=A+B=";X
230  NEXT K
```

```
]RUN
K=1 X=A+B=1948
K=2 X=A+B=974
K=4 X=A+B=487
```

从上述结果可以看出，K符合正整数，且A，B之和也符合正整数的只有三个，故可以确定K的变化范围最小是1，最大是4。

第二，K＝1时，(3)式演变成(2)式。满足题意的解，还应考虑(1)式。这样，才可进一步确定A和B的变化，同样，A、B之值应为正整数，小数无效。

由

$$\begin{cases} A-B=J & (1) \\ A+B=1948 & (2) \end{cases} \quad （K=1时成立）$$

联立解之，则

$$\begin{cases} A=1948/2+J/2 \\ B=1948/2-J/2 \end{cases} \quad (J=1,2,3,4,5,6,7)$$

我们也可以编一个程序，求K＝1时，三位数A、B的变化范围，程序见PROGRAM-THREE。

```
]LIST

295  REM  PROGRAM-THREE
300 K = 1: PRINT "K=";K
305  PRINT
310  FOR J = 1 TO 7
320 A = 1948 / 2 + J / 2
330 B = 1948 / 2 - J / 2
340  IF A =  INT (A) AND B =  INT (B) THEN  PRIN
T "A=";A: PRINT "B=";B
345  PRINT
350  NEXT J
```

```
]RUN
K=1

A=975
B=973
```

```
A=976
B=972

A=977
B=971
```

由上述结果可知，当K＝1时，A和B均为正整数的解也是三组，它们的变化范围：

$A=975-977$，　$B=971-973$

第三，仿照第二种情况，在K＝4时,也应满足(1)式，故有：

$$\begin{cases} A-B=J & (1) \\ A+B=1948/4 & (3) \end{cases}$$ （当K＝4时成立）

联立解之：

$$\begin{cases} A=\dfrac{487+J}{2} \\ \\ B=\dfrac{487-J}{2} \end{cases}$$ （J＝1,2,3,4,5,6,7）

解决上述问题的程序见PROGRAM-FOUR。

```
]LIST

395  REM PROGRAM-FOUR
400 K = 4: PRINT "K=";K
405  PRINT
410  FOR J = 1 TO 7
420 A = (487 + J) / 2
```

```
430 B = (487 - J) / 2
440  IF A =  INT (A) AND B =  INT (B) THEN  PRIN
T "A=";A: PRINT "B=";B
445  PRINT
450  NEXT J
```

```
]RUN
K=4

A=244
B=243

A=245
B=242

A=246
B=241

A=247
B=240
```

由此可知，在K＝4时，A的变化范围为：244—247，B的变化范围为240—243 。

综合以上两种分析结果，A、B变化范围是：

A：244—247 ；　975—977

B：240—243 ；　971—973

由于A、B共同组成6位数密码，故进而确定I的变化范围：240—247；971—977。这样一来，I的变化范围，已

从100—999的 899次，减少到只有14次。而程序ONE的二重循环是899 × 7 =6293次，减少到现在14 × 7 =98次，共减少了6100余次无效循环。

现在，我们来看一下程序 PROGRAM-ONE 的运行结果吧！噢！它已经出来了：

```
]RUN
7-星期日 247240
5-星期五 246241
3-星期三 245242
1-星期一 244243
6-星期六 977971
4-星期四 976972
2-星期二 975973
```

上述程序在PC-1500机上运行了三十分钟左右，在中华学习机上运行五分钟左右，可见速度是相当慢的。但是，我们毕竟找到了“密码”，经逐一投试，果然每天开锁的密码都不相同，但都能打开。

最后，我们想利用前面的分析结果，将程序 PROGRAM-ONE 稍加改动，看看运行时间如何，究竟速度能提高多少？程序见 PROGRAM-FIVE：

```
]LIST

3  REM   破译密码二
5  REM  PROGRAM-FIVE
10  DIM A$(7)
20  FOR K = 1 TO 7
30  READ A$(K)
```

```
40  NEXT K
50  DATA 星期一,星期二,星期三,星期四,星期五,星期
六,星期日
55  INPUT M,N: PRINT
56  PRINT "M=";M;" ";"N=";N
57  PRINT
58  IF M = 1 AND N = 1 THEN  END
60  FOR I = M TO N
70  FOR J = 1 TO 7
80 A = J + I:B = I
90  IF (J / 2 < >  INT (J / 2)) AND (1948 / (A
+ B) =  INT (1948 / (A + B))) THEN  PRINT J;"-";
A$(J);" ";A;" ";B: GOTO 100
100  IF A + B = 1948 THEN  PRINT J;"-";A$(J);" "
;A;" ";B
110  NEXT J
120  NEXT I
125  GOTO 55
128  PRINT
130  END


]RUN
?240,247

M=240 N=247

7-星期日 247 240
5-星期五 246 241
3-星期三 245 242
1-星期一 244 243
?970,977
```

```
M=970 N=977

6-星期六 977 971
4-星期四 976 972
2-星期二 975 973
?1,1

M=1 N=1
```

可见，本程序的结果与第一个程序结果是一致的，而本程序运行时间只有几秒钟，速度比第一个程序提高了100多倍。

本程序运行方法如下：

```
]RUN        ⎫
?240        ⎬ 指 I 的第一种取值范围
??247       ⎭

M=240 N=247

7-星期日 247 240
5-星期五 246 241
3-星期三 245 242
1-星期一 244 243
?970        ⎫
??977       ⎭ 指 I 的第二种取值范围

M=970 N=977
```

```
6-星期六 977 971
4-星期四 976 972
2-星期二 975 973

?1                    } 敲入1,1结束
??1

M=1 N=1
```

本程序结构新颖，解法巧妙，更主要的是利用计算机协助编程。读者可以将上述7组结果，即每天不同的6位码，逐一代入文中开头列出的四个条件中，来验证一下结果的正确性。

二、学 业 管 理

1．找成绩

某个学生，将6门课程名称、学分和成绩，按上述先后次序存放在DATA区中，要求按成绩的等次(A、B、C、D)查找学业情况。试编制一个BASIC程序。

分析：

由于学生按课程名称、学分和成绩的次序已将上述信息存放在DATA语句中，今要求按成绩查找，故可以直接输入成绩等次，用字符串比较的方法处理。

变量设置：A$：课程名称

X：学分

C$：成绩等次

I：循环变量

E$：成绩输入

循环外用INPUT语句输入成绩，循环中先读取DATA区中的信息，如果输入的成绩E$不等于存贮的信息C$，则循环再找；如果相等则打印对应成绩等次的信息。

程序见H13：

说明：请注意循环体外65句 RESTORE 的安排，如果没有这一句，经70句无条件转移上去，再进入循环时将无数可读。这虽是一个简单的问题，但初学者容易忽视，即便是有经验的程序员，稍不留心，也会发生类似的错误。

```
]LIST

2  REM H13
3  REM   找 成 绩
4  REM  课程<A$>,学分<C$>,成绩<E$>
5  DIM A$(15),C$(1),E$(1)
10  INPUT "输入成绩<A,B,C,D>";E$
11  IF E$ = "0" THEN 160
12  PRINT E$
15  PRINT
20  FOR I = 1 TO 6
30  READ A$,X,C$
40  IF E$ < > C$ THEN 60
50  PRINT A$;"---";X;"---";C$
60  NEXT I
65  RESTORE
70  GOTO 10
100  DATA  ENGLISH,4,A
110  DATA  PHYSICS,5,C
120  DATA  CHINESE,5,C
130  DATA  BASIC,3,A
140  DATA  CHEMITY,2,D
150  DATA  LANGUAGE,3,B
160  END
```

程序H13的运行结果如下:

```
]RUN
输入成绩<A,B,C,D>A
A

ENGLISH---4---A
BASIC---3---A
输入成绩<A,B,C,D>B
B

LANGUAGE---3---B
输入成绩<A,B,C,D>C
C

PHYSICS---5---C
CHINESE---5---C
输入成绩<A,B,C,D>D
D

CHEMITY---2---D
输入成绩<A,B,C,D>0
```

讨论：

（1）若本题改为按课程名称查找学业情况，则只要改动40句条件判断语句中的C$ 为A$ 。

（2）若按学分查找学业情况，仅改动40句中的C$ 为X行不行？为什么？请有兴趣的读者不妨上机亲自试一下。

2. 计算机自动出题评分

试编一个程序，要求：

（1）计算机自动出题，每次不一样；

（2）共10道题，每题都包含加、减、乘、除运算；
（3）计算机自动检查学生的答案，打印每题正误；

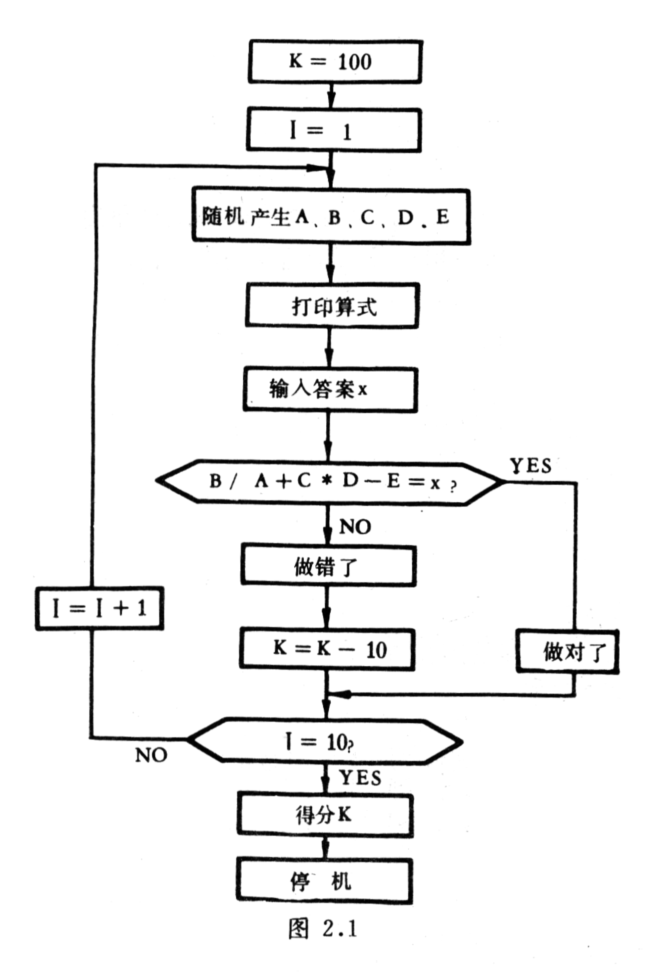


图 2.1

(4) 计算机给出总分数。

设计思路：

利用随机函数RND(X)（其中X可以是任意数值），可以产生一个0到1之间的随机数，再用取整函数INT(X)，可以产生整数。为方便计算可以产生1到2位的随机整数，把它们赋给A、B、C、D、E五个变量，从而产生五个随机数，并可以构成一个包括加、减、乘、除的算式：

$$B/A+C*D-E$$

利用PRINT语句，可以将上述算式打印出来并显示在屏上。小朋友可以用INPUT语句将算式的计算结果送入计算机,再通过条件语句IF/THEN来判断答案的正确和错误，并给出计算是否正确的评语，做错了减10分，即K=K-10，做对了不减，其中K的初值定为100（满分）。

根据以上设计思路，容易画出一个框图，见图2.1。

对照框图2.1，不难写出源程序H14：

```
]LIST

5  REM H14
10  REM 计算机自动出题评分
20 K = 100
30  FOR I = 1 TO 10
35  REM 40-80:随机产生A,B,C,D,E的值
40 A =  INT (10 *  RND (1)) + 1
50 B =  INT (10 *  RND (2)) * A
60 C =  INT (10 *  RND (3))
70 D =  INT (10 *  RND (4))
80 E =  INT (10 *  RND (5))
```

```
90  PRINT "打印算式"
100  PRINT B;"/";A;"+";C;"*";D;"-";E;"="
110  INPUT "输入答案:";X
120  IF B / A + C * D - E = X THEN 170
130  PRINT "IT IS WRONG:做错了"
140  REM 做错1题减10分
150 K = K - 10
160  GOTO 180
170  PRINT "IT IS RIGHT:做对了"
180  NEXT I
185  PRINT
190  PRINT "得分:";" ";K
200  END
```

程序中K为先赋给的总分100，做错一题由总分减去10，X为小朋友根据四则运算题目得出的答案，把它由键盘送入计算机。

下面是程序H14的运行实例：

```
]RUN
打印算式
3/3+6*1-1=
输入答案:6
IT IS RIGHT:做对了
打印算式
21/7+6*5-7=
输入答案:26
IT IS RIGHT:做对了
```

```
打印算式
24/8+6*2-7=
输入答案:8
IT IS RIGHT:做对了
打印算式
16/8+0*7-5=
输入答案:3
IT IS WRONG:做错了
打印算式
16/2+5*9-4=
输入答案:49
IT IS RIGHT:做对了
打印算式
45/9+8*8-8=
输入答案:62
IT IS WRONG:做错了
打印算式
40/8+7*3-4=
输入答案:22
IT IS RIGHT:做对了
打印算式
0/1+7*6-6=
输入答案:36
IT IS RIGHT:做对了
打印算式
60/10+9*2-0=
输入答案:24
IT IS RIGHT:做对了
打印算式
20/10+9*6-9=
输入答案:47
```

```
IT IS RIGHT:做对了
得分: 80
```

3. 课程表

全国各地中小学，正在日益开展计算机普及教育，受到广大青少年喜爱，许多同学积极参加课余计算机学习小组，有的还取得了令人可喜的成绩。特别是中华学习机有处理汉字的功能，更受到广大中小学生的欢迎，用汉字打印报表是一件饶有情趣的事情。下面介绍几个打印课程表的程序。

第一个程序H15，编制十分简单，操作非常简便，对好打印纸，接通打印机电源，按RUN键后，立即打印出一张课程表，见表2.1。课程的内容，可根据学校安排，在DATA语句中输入或修改。下面这个课表（表2.1），用 POKE 1659，5设定：

```
]RUN

  * * * *  课程表   * * * *
   一   二   三   四   五   六
1 数学 物理 语文 数学 语文 物理
2 数学 物理 语文 数学 语文 物理
3 英语 体育 音乐 化学 英语 自习
4 英语 体育 音乐 化学 英语 自习
5 化学 政治 自习 体育 政治 音乐
6 化学 政治 自习 体育 政治 音乐
```

对应表2.1结果的程序清单如程序H15所示：

```
]LIST

1  REM H15
2  PRINT : PRINT
5  REM 课程表
10  DIM A$(6,6)
15  FOR I = 1 TO 6
20  FOR J = 1 TO 6
25  READ A$(I,J)
30  NEXT J
35  NEXT I
40  PRINT  TAB( 4);"* * * *   课程表    * *
* *"
45  PRINT  TAB( 4);"一   二   三   四   五
 六"
50  FOR I = 1 TO 6
55  PRINT I;" ";
60  FOR J = 1 TO 6
65  PRINT A$(I,J);" ";
70  NEXT J
75  PRINT
80  NEXT I
100  DATA 数学,物理,语文,数学,语文,物理
105  DATA 数学,物理,语文,数学,语文,物理
110  DATA 英语,体育,音乐,化学,英语,自习
115  DATA 英语,体育,音乐,化学,英语,自习
120  DATA 化学,政治,自习,体育,政治,音乐
125  DATA 化学,政治,自习,体育,政治,音乐
```

10—35句读存课程信息，40句用 TAB（X）函数定位打印一个表头，45句打印星期，50—80句打印课表内容。请注意打印格式55句，65句，75句的安排，包括标点符号在内，缺一不可。

第二个程序H16，是在程序中设置了打印语句，程序运行后，在屏幕上显示一张课程表，并提示是否需要打印，是的按Y键，反之按任意键结束停机，从而比第一个程序H15增加了更多的灵活性。POKE 1659，0是关闭打印机，它相当于西文状态的PR#O命令。130句 POKE 1659，n，n取值不同时，字型大小改变。程序清单见H16，运行结果(见表2.2)和上面基本相同。A的赋值是一种标志，它可以控制断开打印机和主机的联系。GETK$ 是一个立即执行命令，从键盘中输入一个字符后，不需要按RETURN键。

课程表

	一	二	三	四	五	六
1	数学	物理	语文	数学	语文	物理
2	数学	物理	语文	数学	语文	物理
3	英语	体育	音乐	化学	英语	自习
4	英语	体育	音乐	化学	英语	自习
5	化学	政治	自习	体育	政治	音乐
6	化学	政治	自习	体育	政治	音乐

```
]LIST

1  REM H16
2  PRINT : PRINT
```

```
5  REM "课程表一"
10  DIM A$(6,6)
15  FOR I = 1 TO 6
20  FOR J = 1 TO 6
25  READ A$(I,J)
30  NEXT J
35  NEXT I
40  PRINT  TAB( 14);"课程表"
45  PRINT  TAB( 4);"一    二    三    四    五
 六"
50  FOR I = 1 TO 6
55  PRINT I;" ";
60  FOR J = 1 TO 6
65  PRINT A$(I,J);" ";
70  NEXT J
75  PRINT
80  NEXT I
83  IF A = 1 THEN  POKE 1659,0: END
85  PRINT "是否要打印?(Y/N)"
90  GET K$
95  IF K$ = "Y" THEN 130
96  END
100  DATA 数学,物理,语文,数学,语文,物理
105  DATA 数学,物理,语文,数学,语文,物理
110  DATA 英语,体育,音乐,化学,英语,自习
115  DATA 英语,体育,音乐,化学,英语,自习
120  DATA 化学,政治,自习,体育,政治,音乐
125  DATA 化学,政治,自习,体育,政治,音乐
130  POKE 1659,5
135 A = 1
140  GOTO 40
```

第三个程序H17，是在前面两个程序的基础上，增加了制表功能，从而使输出更为整齐美观。关于制表语句的安排及原理说明，请看“表格程序”。表2.3和表2.4是两个运行实例，用POKE 1659， 1打印的表格字型小一些，用POKE 1659， 5打印的表格字型大一些。

课 程 表

	一	二	三	四	五	六
1	语文	数学	语文	数学	语文	语文
2	数学	语文	数学	语文	数学	数学
3	音乐	语文	语文	语文	美术	体育
4	科技	美术	说话	音乐	文艺	阅读
5	思品	活动	体育	活动	自习	写字
6	——	——	——	——	——	班会

南京市　北京路小学二年级　朱伟伟

```
]RUN
```

课 程 表

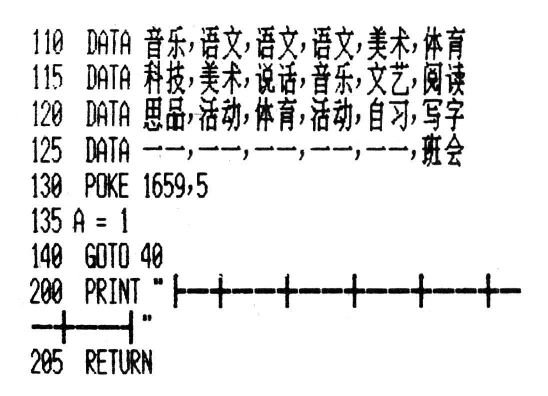
	一	二	三	四	五	六
1	语文	数学	语文	数学	语文	语文
2	数学	语文	数学	语文	数学	数学
3	音乐	语文	语文	语文	美术	体育
4	科技	美术	说话	音乐	文艺	阅读
5	思品	活动	体育	活动	自习	写字
6	——	——	——	——	——	班会

南京市 北京路小学二年级 朱伟伟

```
]LIST

1  REM H17
2  PRINT : PRINT : PRINT
5  REM  课程表2
10  DIM A$(6,6)
15  FOR I = 1 TO 6
```

```
20  FOR J = 1 TO 6
25  READ A$(I,J)
30  NEXT J
35  NEXT I
40  PRINT  TAB( 15);"课  程  表"
41  IF A = 1 THEN  POKE 1659,1
43  PRINT "┌──┬────┬────┬────┬────┬────┬
─┬───┐"
45  PRINT "│  │ 一 │ 二 │ 三 │ 四 │ 五
 │ 六 │"
46  GOSUB 200
50  FOR I = 1 TO 6
55  PRINT "│";I;" ";"│";
60  FOR J = 1 TO 6
65  PRINT A$(I,J);"│";
70  NEXT J
75  PRINT
76  IF I = 6 THEN  PRINT "└──┴────┴────┴
────┴────┴────┴────┘": GOTO 80
78  GOSUB 200
80  NEXT I
82  PRINT "       南京市   北京路小学二年级
朱伟伟"
83  IF A = 1 THEN  POKE 1659,0: END
84  PRINT : PRINT
85  PRINT "是否要打印?(Y/N)"
90  GET K$
95  IF K$ = "Y" THEN 130
96  END
100  DATA 语文,数学,语文,数学,语文,语文
105  DATA 数学,语文,数学,语文,数学,数学
```



```
110  DATA 音乐,语文,语文,语文,美术,体育
115  DATA 科技,美术,说话,音乐,文艺,阅读
120  DATA 思品,活动,体育,活动,自习,写字
125  DATA 一一,一一,一一,一一,一一,班会
130  POKE 1659,5
135 A = 1
140  GOTO 40
200  PRINT "├─┼──┼──┼──┼──┼──┼─
─┼──┤"
205  RETURN
```

4. 表格程序

编制一个程序，要求能制成如表2.5格式的报表：

]RUN

	一	二	三
1	110	120	130
2	140	150	160
3	200	210	220
4	230	240	250

上面这个报表在实际工作中常常可以看到，显然，能够绘制表2.5报表，则一般类似的报表也可以画出来。所以我们在编制这个报表程序时，尽可能把方法叙述得清楚些，以供使用者学习和借鉴，并可以以本程序为模式，编制出更多更好的报表来。

分析：

(1) 如果不考虑一，二，三的部分，表的下半部是四行三例，而表中放的是数值型数据，因此，用二组数组二重循环处理读存数据是比较方便的，变量取双下标变量A（I，J）比较适合，I＝1，2，3，4，J＝1，2，3。而所有数据放在DATA语句中，因此，在程序的开始首先把数据存贮起来，以备填表时使用。这项工作由2—210句完成，程序段见H18：

```
]LIST1,210

1  REM H18
2  REM  表格程序
5  REM   A(4,3)
10  DIM A(4,3)
20  FOR I = 1 TO 4
30  FOR J = 1 TO 3
40  READ A(I,J)
60  NEXT J
80  NEXT I
200  DATA 110,120,130,140,150,160
210  DATA 200,210,220,230,240,250
```

(2) 关于表的上半部，即包含中文数字一、二、三的部份，可把它作为一个表头处理。中华学习机有制表符号，利

用它们确实是很方便的，只是要注意符号的正确选用和打印格式的对齐处理。在中文拼音状态下，按 \ ↾ 键可以灵活选用制表符，配合 < . ，> , 两个键，原则上可以满足不同的制表要求。完成表头制作只需要安排300，310两句。

```
]LIST300,310

300  PRINT "┌──┬──┬──┬──┐"
310  PRINT "│  │ 一 │ 二 │ 三 │"
```

（3）紧接着表头以后，要画一根横线，而为了连接表的下半部，又常常需要画"|———|———|"线，故安排了400句，它由320句调用，这实际上是调子程序，而子程序又来得特别简单，在需要画线的地方，正确调用，完成画线任务后再自动返回，如：

```
]LIST

320  GOSUB 400
400  PRINT "├──┼──┼──┼──┤"
410  RETURN
```

这样处理是因为制表中不止一次要用到如上的画线，而用子程序的方法使程序更为简洁方便。程序中392句的功能同上。

（4）为了完成下半部制表任务，请注意它不仅要画线，对齐，还要写数据，这项工作由330—394句的程序段完成。

请注意二重循环中语句、格式以及制表符号的安排。例如340,360,380三句的处理。另外390句是在完成全部报表后要加的一个报尾。本段程序是整个制表工作的中心，格式很有讲究，几乎缺一不可。程序段见H19:

```
]LIST325,396

325  REM H19
330  FOR I = 1 TO 4
340  PRINT "|";I;"  ";"|";
350  FOR J = 1 TO 3
360  PRINT A(I,J);"  |";
370  NEXT J
380  PRINT
390  IF I = 4 THEN  PRINT "└──┴────┴───
┴───┘": GOTO 394
392  GOSUB 400
394  NEXT I
396  END
```

说明:

本程序基本上是一个打印表格程序的模型，类似的报表程序原则上照此办理，如果数据更多，可以改变循环的终值，相应的数据以及增加画线格式的长度。完整的程序清单（见H20）及运行结果:

```
]RUN
```

	一	二	三
1	110	120	130
2	140	150	160
3	200	210	220
4	230	240	250

```
]LIST

1  REM  H20
2  REM  表格程序
5  REM   A(4,3)
10  DIM A(4,3)
20  FOR I = 1 TO 4
30  FOR J = 1 TO 3
40  READ A(I,J)
60  NEXT J
80  NEXT I
200  DATA 110,120,130,140,150,160
210  DATA 200,210,220,230,240,250
300  PRINT "┌───┬───┬───┬───┐"
310  PRINT "│   │ 一 │ 二 │ 三 │"
320  GOSUB 400
```

```
330 FOR I = 1 TO 4
340 PRINT "|";I;" ";"|";
350 FOR J = 1 TO 3
360 PRINT A(I,J);" |";
370 NEXT J
380 PRINT
390 IF I = 4 THEN  PRINT "└───┴───┴───
┴───┘": GOTO 394
392 GOSUB 400
394 NEXT I
396 END
400 PRINT "├───┼───┼───┼───┤"
410 RETURN
```

5. 通讯录

通讯录编制方法比较简单，首先确定有关信息的名称，例如姓名、性别、城市、地址、邮政号码、电话号码等，并把它们分别存放在字符串变量A$，B$，C$，D$，E$和F$中，然后用FOR /NEXT循环来读取并打印。

当然，这些信息的存贮，也并不一定都要用字符串变量，例如邮政编码全是数字常数，可以用简单变量（如X）来存放。但中文汉字必须用字符串变量来存贮。

READ和DATA语句必须配对，在从键盘敲入DATA区的信息时，应该特别小心，一个标点符号的失误，将会使许多信息出现错误。

DATA区中的字符串信息，按基本BASIC规定，必须加引号，如“王申”等，但在 APPLE 机和中华学习机

以及相兼容的紫金Ⅱ中，引号可以省去。有些微机，例如PC-1500，使用DATA语句时，字符串必须加引号。

为了使通讯录具有查找的灵活性，可以安排一个查找的程序段，例如输入一个数值I，打印出对应于I的串变量(如A$(I)，B$(I)，…)的值。

根据上述分析，我们容易写出通讯录程序H21：

```
]LIST

2  REM H21
5  REM 通  讯  录
7  REM    "开 辟 数 组"
8  REM 姓名(A$),性别(B$),城市(C$),地址(D$),
邮政编码(E$),电话号码(F$)"
10  DIM A$(20),B$(9),C$(20),D$(40),E$(5),F$(
12)
20  FOR I = 1 TO 5
30  READ A$(I),B$(I),C$(I),D$(I),E$(I),F$(I)
40  PRINT I;" ";A$(I);" ";B$(I);" "
50  PRINT C$(I);" ";D$(I);" ";
60  PRINT E$(I);" ";F$(I)
65  PRINT
70  NEXT
80  DATA 王 申,男,北京,韶山路145号,13421,658
425-376"
90  DATA  程 宏,男,南京,北京西路214号,100345
,634651-281"
100  DATA 李 铭,男,天津,上海西子路868号,2342
51,776821-45"
110  DATA 常华,男,北京,王府井7号,123235,7723
51-33
```

```
120  DATA 雷 磊,男,西安,浙江路535号,65331,10
0421-56"
130  PRINT : PRINT
140  INPUT "I=";I
150  IF I =  - 1 THEN 200
160  PRINT A$(I);" ";B$(I);" ";C$(I);" ";
170  PRINT D$(I);" ";E$(I);" ";F$(I);
175  PRINT
180  GOTO 140
200  END
```

可见编制通讯录程序的方法比较简单，其主要思路是：

- 开辟一定的数组；
- 用循环读取信息；
- DATA语句存放记录；
- 用打印语句输出；
- 用简单方法查找。

变量设置：

A$(I)——存放姓名；

B$(I)——存放性别；

C$(I)——存放城市名称；

D$(I)——存放地址；

E$(I)——存放邮政编码；

F$(I)——存放电话号码；

I——循环控制变量。

几点说明：

· 10句是开辟数组，本程序以5个通讯录为例，名字符串变量的下标要合适，例如它们都不得小于5，否则出错。

如果存贮信息较多，则数组下标应大于存贮信息的数目。

·20—70句是一个循环，目的是自动循环读数，由于采用字符串变量，则经过循环后，各种信息就存贮在相应的字符串变量中，例如 A$(3)中存放李铭这两个字，这样处理为检索为准备。循环变量的终值应根据实际存贮信息的个数而定，本例是5。

·为使程序更加灵活，可以加一句15，即 ONERR GOTO 130，这样安排好处很多，一是DATA区的信息可以随意增减，可以随时增加信息，也可以随时删除记录，但终值可以取大一些，例如100；二是防止出错，例如终值取100后，信息可以少于100，而不会发生无数可读的情况。

·本例的检索（又称查找）方法最为简单，利用140句键盘输入一个数（本例是1,2,3,4,5），就可以查找到顺序存贮在 DATA 语句中的信息，而I取－1，则结束本程序，这里I＝－1就是程序的终止标志。

下面是程序运行的实例：

```
]RUN
1 王 申 男
北京 韶山路145号 13421 658425-376"

2 程 宏 男
南京 北京西路214号 100345 634651-281"

3 李 铭 男
天津 上海西子路868号 234251 776821-45"

4 常华 男
北京 王府井7号 123235 772351-33

5 雷 磊 男
```

```
西安 浙江路535号 65331 100421-56"


I=1
王 申 男 北京 韶山路145号 13421 658425-376"
I=3
李 铭 男 天津 上海西子路868号 234251 776821-
45"
I=5
雷 磊 男 西安 浙江路535号 65331 100421-56"
I=4
常华 男 北京 王府井7号 123235 772351-33
I=2
程 宏 男 南京 北京西路214号 100345 634651-28
1"
I=-1
```

上面介绍的通讯录程序，检索方法简单但不够灵活，同时程序编制比较呆板，没有创新。现在，我们再列出一个通讯录程序（见H 21—11)，并列出它的部份结果。关于这个程序的详细说明，我们把它放在情报检索一节中介绍。应该指出的是30—240句是一个非常有用的子程序。

```
]LIST

10  REM H21-11
20  REM  SEARCHING PROGRAM
30  DIM A$(10)
40  ONERR  GOTO 100
50  FOR J = 1 TO 10
60  READ A$(J)
70  NEXT J
```

```
80  PRINT "--------------------------------
-------"
90  PRINT
100  INPUT B$
110  IF B$ = "#" THEN 240
120 N = 0
130  FOR J = 1 TO 10
140 N = N + 1
150  FOR I = 1 TO  LEN (A$(J)) -  LEN (B$) +
 1
160  IF B$ < > MID$ (A$(J),I, LEN (B$)) TH
EN 190
170  PRINT B$;" ";"No:";" ";N
180  PRINT "*";A$(J);"*"
190  NEXT I
200  NEXT J
210  RESTORE
220  PRINT "--------------------------------
--------"
230  GOTO 100
240  END
250  DATA  王 申 男 北京 韶山路145号 13421 6
58425-376
270  DATA 程 宏 男 南京 北京西路214号 100345
 634651-281
280  DATA  李 铭 男 天津 上海西子路868号 234
251 776821-45
290  DATA 常华 男 北京 王府井7号 123235 7723
51-33
300  DATA  雷 磊 男 西安 浙江路535号 65331 1
00421-56
```

```
]RUN
?-33
-33 No: 4
*常华 男 北京 王府井7号 123235 772351-33  *
--------------------------------------------
?京
京 No: 1
*王 申 男 北京 韶山路145号 13421 658425-376
*
京 No: 2
*程 宏 男 南京 北京西路214号 100345 634651-2
81  *
京 No: 2
*程 宏 男 南京 北京西路214号 100345 634651-2
81  *
京 No: 4
*常华 男 北京 王府井7号 123235 772351-33  *
--------------------------------------------
?1003
1003 No: 2
*程 宏 男 南京 北京西路214号 100345 634651-2
81  *
--------------------------------------------
?100
100 No: 2
*程 宏 男 南京 北京西路214号 100345 634651-2
81  *
100 No: 5
*雷 磊 男 西安 浙江路535号 65331 100421-56 *
--------------------------------------------
```

```
?535
535 No: 5
*雷 磊 男 西安 浙江路535号 65331 100421-56 *
-----------------------------------------
?#
```

运行结果表明，该程序检索方法灵活多样，既可以输入汉字也可以输入数码（实际上如果 DATA 语句中存放英语或汉语拼音也可以查找）；既可以输入几个字符也可以仅仅输入一个字符，都可以查找到有关信息。

例如，程序运行后出现？在中文状态敲入一个“京”字，它就可以同时查找到籍贯为北京，南京的所有人的信息。

又如，一个名叫常华的人，他的电话号码为772351-33，而要查找他的情况，只要输入-33，即可，-33是772351-33中的几个字符。

不仅如此，本程序可以任意查找，例如，只记得雷磊同志家庭地址是535号，那么输入535号，即可查找他的全部通讯录内容。

读者可以反复运行本程序，用各种最简单的方法进行查找，从而体会本程序的许多优点。

特别是改动 DATA 区信息后，本程序可以用这一切情报检索的地方。

6. 日历程序

设计一个程序，打印1988年日历表，要求按表 2.6 和表 2.7两种形式打印：

表 2.6

```
]RUN


                    <--1988-->

                       <1>
 0      1      2      3      4      5      6
SUN    MON    TUE    WED    THU    FRI    SAT
                                    1      2
 3      4      5      6      7      8      9
 10     11     12     13     14     15     16
 17     18     19     20     21     22     23
 24     25     26     27     28     29     30
 31

                       <2>
 0      1      2      3      4      5      6
SUN    MON    TUE    WED    THU    FRI    SAT
        1      2      3      4      5      6
 7      8      9      10     11     12     13
 14     15     16     17     18     19     20
 21     22     23     24     25     26     27
 28     29
```

表 2.7

```
                          <--1988-->

                             <1>
星期日    星期一    星期二    星期三    星期四    星期五    星期六
 SUN       MON       TUE       WED       THU       FRI       SAT
                                                    1         2
  3         4         5         6         7         8         9
  10        11        12        13        14        15        16
  17        18        19        20        21        22        23
  24        25        26        27        28        29        30
  31

                             <2>
星期日    星期一    星期二    星期三    星期四    星期五    星期六
 SUN       MON       TUE       WED       THU       FRI       SAT
            1         2         3         4         5         6
  7         8         9         10        11        12        13
  14        15        16        17        18        19        20
  21        22        23        24        25        26        27
  28        29
```

分析：

我们首先分析第一种日历表的设计方法。

（1）打印1988年表头，这个工作最为简单，用字符串原样照印语句，再配合定位的TAB（X）函数，如：

PRINT TAB(15)；"<－－1988－－>"

（2）设置1988年元月一日是星期几的语句，因为打印一个日历，总是从元旦这一天开始的，如1988年1月1日是星期五，可以用赋值语句N＝5处理。也可以安排一个INPUT语句，输入任意一年元旦是星期几的数字，这样可以打印任一年日历。

（3）为简化程序设计，省去对润年的判断，例如，1988年是润年，故2月份应为29天，将此数据放在DATA语句中。其它月份的天数也按实际月份的天数处理。即DATA 31，29，31，30，31，30，31，31，30，31，30，31。按1到12月的天数顺序存放。

（4）印每月日历表的表头，包括三项内容：注明月份；列印星期（用0，1，2，3，4，5，6代替）；打印对应的英文缩写（如星期日为SUN，星期一为MON，……，星期六为SAT）。这三项内容均可用PRINT语句和TAB函数来定位。考虑到这几项内容在每月都要打印，显然放在月份的循环语句中比较方便，用工作为循环控制变量，记取月份（1到12）。如：

```
]LIST30,70

30  FOR I = 1 TO 12
40  PRINT
50  PRINT  TAB( 19);"(";I;")";: PRINT
```

```
60 PRINT "   0   1   2   3   4   5   6"
70 PRINT "  SUN MON TUE WED THU FRI SAT"
     :
     :
     :
     :
     :
180 NEXT I
```

(5) 印每月日历表，这是整个程序设计的关键。用一个循环来控制，循环控制变量取J，它是记日数的，容易想到循环总是从每月第一天开始，到该月的最后一天结束，因此循环变量的初值为1，终值为M，M是一个工作单元，分别为各月的天数。为使程序自动运行，天数的读取用 READ 语句处理，显然 READ M应放在外循环之中（I 循环），且安置在内循环之外（J循环）。

为了在相应的位置上列印日期数字，定位是至关重要的，特别是元月一日这一天应放在星期五（FRI）下面的位置上，经计算用PRINT TAB(5 * (N + 1)); J; 语句处理，为了使其它日期也能正确定位，程序中安排了一个计数器：N = N + 1，只要N <= 6 就再循环打印，并不断计数，而当N > 6 时，则应让计数器置零（N = 0）并换行，这样，就能使星期日打印正确的数值。当一个月所有天数都打印完毕后，还应使计数器置零。列印日期的主要程序段为：

```
]LIST100,140

100 FOR J = 1 TO M
110 PRINT  TAB( 5 * (N + 1));J;
```

```
120 N = N + 1
130  IF N <  = 6 THEN 150
140 N = 0: PRINT
```

5 NEXT J

(6) 程序结构和变量说明，整个程序用两重循环进行，外循环读月份，用 I 做外循环控制变量；内循环记日数，用 J 作内循环控制变量。N 既是记录某日在一周内的序号，表示星期几，又作为计数器。M 为每月天数，它供作内循环的终值。

至此，整个程序设计完成，清单见 H22：

```
]LIST

2  REM H22
5  PRINT : PRINT
10  PRINT  TAB( 15);"(--1988--)"
20 N = 5
30  FOR I = 1 TO 12
40  PRINT
50  PRINT  TAB( 19);"(";I;")";: PRINT
60  PRINT "    0    1    2    3    4    5    6"
70  PRINT "   SUN  MON  TUE  WED  THU  FRI  SAT"
80  READ M
90  DATA 31,29,31,30,31,30,31,31,30,31,30,31
100  FOR J = 1 TO M
110  PRINT  TAB( 5 * (N + 1));J;
120 N = N + 1
130  IF N <  = 6 THEN 150
140 N = 0: PRINT
150  NEXT J
```

```
160  IF N = 0 THEN 180
170  PRINT
180  NEXT I
190  END
```

现在，我们再来分析第二种格式的日历程序设计。

从表面上看，只要将60句的“0”，“1”，……，“6”改成“星期日”，“星期一”，……，“星期六”。实际上并不那么简单。主要原因在于换成汉字信息后，日期不能正确对齐。

中华学习机在西文和中文状态下，打印格式有很大的差别：在英文状态下，一个字符占一个格子（一个字节）；而中文状态下，一个汉字占两个格子（二个字节）。这样，在中英文混合状态时，用TAB（X）函数定位不易处理。再说，中文状态下，一般每行处理17个汉字，或者说34个字符，因此是以34为模的，因此第一种格式的日历程序要作适当改动。

·元月一日放在X＝7＊（N＋1）－A的位置上，开始A＝0，N＝5，所以X＝42，这个位置定的正确，TAB函数第一次超过34时，可以执行。它是定在星期五（N＝5）的下面。计数器加1后W＝6，此时应安排A＝34，循环上去后X＝7＊（N＋1）－A＝7＊（6＋1）－34＝49－34＝15，TAB（X）＝TAB（15），就打印在星期六的下面。所以在程序中要安排条件语句：IF N＞＝5 THEN A＝34。

·对星期日的对应日期处理，应在N＞6以后，设置N＝0：A＝0：PRINT，这样处理是因为星期日（N＝0）总是在表的最左边，它是第7个位置上，故X＝7＊（N＋1）－A＝7＊1－0＝7。

所以，为了在中英文混合状态下，正确打印文中开头提到的第二种格式，必须在第一个程序的基础上，加以下条件及赋值语句，它们是：

```
]LIST103,140

103  IF J = 1 THEN A = 0
105 X = 7 * (N + 1) - A
125  IF N >  = 5 THEN A = 34
130  IF N <  = 6 THEN 150
140 N = 0:A = 0: PRINT
```

第二种日历打印格式的程序清单（见H23）和全部运行结果：

```
]LIST

2  REM H23
4  REM  日历程序1
5  PRINT : PRINT
10  PRINT  TAB( 22);"(--1988--)"
20 N = 5
30  FOR I = 1 TO 12
40  PRINT
50  PRINT  TAB( 26);"(";I;")";: PRINT
60  PRINT "   星期日 星期一 星期二 星期三 星期四 星期五
星期六"
70  PRINT "     SUN    MON    TUE    WED    THU    FRI
  SAT"
80  READ M
90  DATA 31,29,31,30,31,30,31,31,30,31,30,31
100  FOR J = 1 TO M
103  IF J = 1 THEN A = 0
```

```
105 X = 7 * (N + 1) - A
110  PRINT  TAB( X);J;
120 N = N + 1
125  IF N >  = 5 THEN A = 34
130  IF N <  = 6 THEN 150
140 N = 0:A = 0: PRINT
150  NEXT J
160  IF N = 0 THEN 180
170  PRINT
180  NEXT I
190  END
```

```
]RUN
                          (--1988--)

                            (1)
星期日   星期一   星期二   星期三   星期四   星期五   星期六
 SUN      MON      TUE      WED      THU      FRI      SAT
                                               1        2
  3        4        5        6        7        8        9
  10       11       12       13       14       15       16
  17       18       19       20       21       22       23
  24       25       26       27       28       29       30
  31

                            (2)
星期日   星期一   星期二   星期三   星期四   星期五   星期六
 SUN      MON      TUE      WED      THU      FRI      SAT
           1        2        3        4        5        6
  7        8        9        10       11       12       13
  14       15       16       17       18       19       20
  21       22       23       24       25       26       27
  28       29
```

(3)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

(4)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

(5)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

(6)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

(7)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

(8)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

(9)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

(10)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

(11)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

(12)

星期日 SUN	星期一 MON	星期二 TUE	星期三 WED	星期四 THU	星期五 FRI	星期六 SAT
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

7. 分数段

这是一个多功能的检索统计程序，以查找学生考分为例，主要功能如下：

·打印全部考生信息。包括准考证号码（NO），姓名(Name)，6门课成绩（为简单计6门课程分别用1，2，3，4，5，6作为代号），各门课程总分（TOL），平均成绩（AVE）以及名次（ORDER）。程序运行后，即可以打印全部考生（本例为8人）考试情况的报表。

·有菜单提示，它既是本程序操作方法的说明，也是本程序检索功能的说明，主要有：

（1）输入准考证号码（NUMBER），查找该生全部信息，包括准考证号码、姓名、各科成绩、总分、平均分。

（2）输入姓名（NAME），查找该姓名考生的成绩情况。

（3）输入准考证号码（NO），查找该生总分。

（4）输入分数（S），查找超过该分数人员的情况，包括准考证号码、姓名和总分，并统计出大于等于该分数的人数。此项功能特别适用于招考学校确定分数线的依据。

（5）输入5，计算机列出各分数档人员数，如400—499分有5人，500—599分有3人等等。

（6）输入6，结束。

·全部程序实现总控，可以灵活地进行各种功能的检索，既可以顺序查找，也可以前后任意搜索，对于数据区中没有的信息，会显示有关信息，如"WITHOUT"没有，"NOT FOUND"找不到等，并可以自动返回总控再找，总控的提

示信息为“#”。

·所有检索均有中文提示，这给管理人员带来方便。

程序分段说明：

·程序运行后，出现?，此时从键盘输入8，6↙，即可打印一个统计表（TON Ji Biao)。

·信息存贮，由8—110句完成，采用两重循环，外循环读取准考证号码（N(I)），考生姓名（A$(I)），内循环读取考生成绩，放在A（I,J)中。

·排序，由200—280句完成，是以总分高低，从大到小排列，凡是不满足这个条件的，就交换,注意准考证号码、姓名、各科成绩、总分、平均分都要同时交换，否则张冠李戴，排序是采用枚举法。

·打印统计表，由285—465句完成，全部格式按西文要求处理。

·菜单提示，由590—612句完成。

·总控，由620—650句完成。

·输入准考证号码查找该生各科考分，由700—790句完成。

·输入姓名查找该姓名考生成绩情况，由800—895句完成。

·输入准考证号码查找总分，由900—945句完成。

·输入分数查找超过该分数人员情况，由1000—1080句完成。

·键盘输入5↙，计算机自动列出各分数段人员数目，由1110—1260句完成。

·键盘输入6↙，执行650句结束。

程序中的中文提示间接地说明了程序的流程，又便于程序管理。

运行实例和完整的程序清单（见H24）；

```
]RUN
?8,6
A                          --------Ton Ji Biao--------

!--------------------------------------------------------------------------------!
! No ! Name  !  1  !  2  !  3  !  4  !  5  !  6  !  TOL  !  AVE  ! ORDER!
!--------------------------------------------------------------------------------!
! 239! SMITH !90   ! 93  ! 87  ! 89  ! 95  ! 87  ! 541  ! 90  ! 1

!--------------------------------------------------------------------------------!
! 721! CHANG !83   ! 67  ! 96  ! 95  ! 87  ! 98  ! 526  ! 88  ! 2

!--------------------------------------------------------------------------------!
! 947! HEIEN !95   ! 97  ! 98  ! 67  ! 75  ! 89  ! 521  ! 87  ! 3

!--------------------------------------------------------------------------------!
! 354! CHENG !94   ! 81  ! 82  ! 73  ! 59  ! 91  ! 480  ! 80  ! 4

!--------------------------------------------------------------------------------!
! 471! JIANG !87   ! 93  ! 45  ! 94  ! 69  ! 74  ! 462  ! 77  ! 5

!--------------------------------------------------------------------------------!
! 892! SHEMO !88   ! 77  ! 65  ! 43  ! 91  ! 90  ! 454  ! 76  ! 6

!--------------------------------------------------------------------------------!
! 652! CHIEN !66   ! 68  ! 87  ! 81  ! 57  ! 93  ! 452  ! 75  ! 7

!--------------------------------------------------------------------------------!
! 453! JONES !66   ! 69  ! 81  ! 92  ! 60  ! 83  ! 451  ! 75  ! 8

!--------------------------------------------------------------------------------!
A!  TONG JI    ! 669 ! 645 ! 641 ! 634 ! 593 ! 705 ! 3887 ! 648  !
!--------------------------------------------------------------------------------!
```

```
1.INPUT NUMBER:输入准考证号码找该生各科考分
2.INPUT NAME:输入姓名
3.INPUT NO:输入准考证号码找总分
4.INPUT S:输入分数找超过该分数人员情况
5.INPUT SCDRE 输入分数段
6.输入6结束

#1
NUMBER:输入准考证号码找该生各科考分 721
721 CHANG 83 67 96 95 87 98 526 88
```

```
#2
 NAME:输入姓名SHEMO
SHEMO 892 88 77 65 43 91 90 454 76


#3
NO:输入准考证号码找总分 453
453---451


#4
 S:输入分数找超过该分数人员情况 450
                  239 SMITH 541
                  721 CHANG 526
                  947 HEIEN 521
                  354 CHENG 480
                  471 JIANG 462
                  892 SHEMO 454
                  652 CHIEN 452
                  453 JONES 451

>=450:8


#5
 分数段
100---199 0
200---299 0
300---399 0
400---499 5
500---599 3
600---699 0


#6
END


]LIST 2,240

2 REM H24
5 REM 分 数 段
8 CLEAR
9 REM 输 入 8 ,6
```

```
10  INPUT N,M
15  REM  A(I,J):放信息
20  DIM A(N + 2,M + 4)
25  REM   N(I):放准考证号码,A$(I):放姓
 名
30  FOR I = 1 TO N
50  READ N(I),A$(I)
60  FOR J = 1 TO M
70  READ A(I,J)
80 A(I,M + 1) = A(I,M + 1) + A(I,J)
90 A(I,M + 2) =  INT (A(I,M + 1) / M + 0.5)
100  NEXT
110  NEXT
190  REM  排序
200  FOR I = 1 TO N - 1
210  FOR J = I + 1 TO N
220  IF A(I,M + 1) > A(J,M + 1) THEN 270
230 T = A(I,M + 1):A(I,M + 1) = A(J,M + 1):A
(J,M + 1) = T
232 F = A(I,M + 2):A(I,M + 2) = A(J,M + 2):A
(J,M + 2) = F
235 A = N(I):N(I) = N(J):N(J) = A
240 A$ = A$(I):A$(I) = A$(J):A$(J) = A$
245  FOR X = 1 TO M
250 C = A(I,X):A(I,X) = A(J,X):A(J,X) = C
260  NEXT
270  NEXT
280  NEXT
285  PRINT  CHR$ (27) + "A" +  CHR$ (5);
288  POKE 1657,80
290  PRINT  TAB( 26);"--------Ton Ji Biao---
```

```
-----": PRINT
291  GOSUB 450
293  PRINT "! No ! Name  !  1  !  2   !  3
 !  4   !  5   !  6   !  TOL  !  AVE  ! ORDE
R!"
296  GOSUB 450
300  FOR I = 1 TO N
305  POKE 1529,255
310  PRINT "! ";N(I);"!";" ";A$(I);" !";
320  FOR J = 1 TO M + 2
330  PRINT A(I,J);"   ! ";
340  NEXT
350  PRINT I
355  PRINT
356  POKE 1529,0: PRINT
358  GOSUB 450
360  NEXT
370  PRINT  CHR$ (27) + "A" +  CHR$ (5);
375  PRINT "!  TONG JI   !";
400  FOR J = 1 TO M + 2
410  FOR I = 1 TO N
420 A(N + 1,J) = A(N + 1,J) + A(I,J)
430  NEXT
440  PRINT " ";A(N + 1,J);"  !";: NEXT
445  PRINT : GOSUB 450: GOTO 465
450  PRINT "!------------------------------
-----------------------------------------
---!"
455  RETURN
465  PRINT  CHR$ (27) + "A" +  CHR$ (12);
500  DATA  471,JIANG,87,93,45,94,69,74
```

```
510  DATA  354,CHENG,94,81,82,73,59,91
520  DATA  721,CHANG,83,67,96,95,87,98
530  DATA  652,CHIEN,66,68,87,81,57,93
540  DATA  892,SHEMO,88,77,65,43,91,90
550  DATA  239,SMITH,90,93,87,89,95,87
560  DATA  453,JONES,66,69,81,92,60,83
570  DATA  947,HEIEN,95,97,98,67,75,89
580  PRINT : PRINT
590  PRINT "1.INPUT NUMBER:输入准考证号码找
该生各科考分"
595  PRINT "2.INPUT NAME:输入姓名"
600  PRINT "3.INPUT NO:输入准考证号码找总分"
605  PRINT "4.INPUT S:输入分数找超过该分数人
员情况"
610  PRINT "5.INPUT SCDRE 输入分数段"
612  PRINT "6.输入6结束"
615  PRINT : PRINT
620  INPUT "#";W
630  ON W GOSUB 700,800,900,1000,1110,650
640  GOTO 620
650  PRINT "END": END
700  INPUT "NUMBER:输入准考证号码找该生各科
考分";P
710  FOR I = 1 TO N
720  IF N(I) = P THEN  PRINT N(I);" ";A$(I);
" ";: GOTO 750
730  NEXT
740  PRINT "WITHOUT!";" ";P: PRINT : GOTO 79
0
750  FOR J = 1 TO M + 2
760  PRINT A(I,J);" ";
```

```
770  NEXT
780  PRINT
790  RETURN
800  INPUT " NAME:输入姓名";K$
810  FOR I = 1 TO N
820  IF A$(I) = K$ THEN  PRINT A$(I);" ";N(I
);" ";: GOSUB 840
830  GOTO 890
840  FOR J = 1 TO M + 2
850  PRINT A(I,J);" ";
860  NEXT
870  PRINT
880  RETURN
890  NEXT
895  PRINT : RETURN
900  INPUT "NO:输入准考证号码找总分";C
910  FOR I = 1 TO N
920  IF N(I) = C THEN  PRINT N(I);"---";A(I,
7): GOTO 945
930  NEXT
940  PRINT "NOT FOUND!";C
945  RETURN
1000 K = 0: INPUT " S:输入分数找超过该分数人
员情况";S
1010  FOR I = 1 TO N
1030  IF A(I,7) >  = S THEN K = K + 1: PRINT
 TAB( 15);N(I);" ";A$(I);" ";A(I,7)
1040  NEXT
1045  PRINT
1070  PRINT ">=";S;":";K
1080  RETURN
```

```
1110  PRINT " 分数段"
1115 B1 = 0:B2 = 0:B3 = 0:B4 = 0:B5 = 0:B6 =
 0
1120  FOR I = 1 TO N
1130 Q =  INT (A(I,7) / 100)
1140  ON Q GOSUB 1155,1160,1170,1180,1190,11
95
1150  NEXT
1153  GOTO 1200
1155 B1 = B1 + 1: RETURN
1160 B2 = B2 + 1: RETURN
1170 B3 = B3 + 1: RETURN
1180 B4 = B4 + 1: RETURN
1190 B5 = B5 + 1: RETURN
1195 B6 = B6 + 1: RETURN
1200  PRINT "100---199";" ";B1
1210  PRINT "200---299";" ";B2
1220  PRINT "300---399";" ";B3
1230  PRINT "400---499";" ";B4
1240  PRINT "500---599";" ";B5
1250  PRINT "600---699";" ";B6
1260  RETURN
```

8. 评语程序

这是一个专门为中小学班主任老师编写的程序，它可以在很短的时间内完成对学生评语的草拟工作，并可以直接打印出来，大大地减少了书写评语的工作量，受到了许多老师的普遍欢迎。

程序编制非常简短，使用也极其方便，即使一个完全不懂计算机的教师，只要稍作说明即可运用自如，工作效率也很高，在一个小时左右的时间内，可以顺利完成对40—50名学生的评语草拟工作。

程序简要说明：

(1) 5—40句为第一段，用循环的方法读存评语信息，评语共用了40句短语，存放在数组A$(I)中，每个评语冠以代号，作为老师在做评语时的依据。35句的作用就是在程序运行后打印出所有评语和代号，实际工作时，只要打印出一张评语代号表后，保存起来，以便以后使用，而可以删去35句。如果作为表演程序，可以保留35句，它实际上是一个评语菜单提示。

(2) 50—120句为程序的第二段，也是本程序的关键程序段。这段程序要求老师将对每一个学生的评语，用相应的代号存入300句开始的DATA语句中，每一个学生用一个DATA语句，而在DATA语句的结尾放一个“0”标记，当所有学生评语代号存放完毕后，最后一个学生的DATA语句结尾放“-1”标记，表示结束。程序运行后，全部学生的评语均可显示在屏幕上，也可全部分别打印出来。

(3) 程序中有几句安排技巧，一是第2句安排了空二行，

目的是将打印纸上]RUN 和打印结果空开来，保证打印结果美观；二是90句也安排了三个PRINT语句，这样使前后两个学生的评语空开三行，以便老师整理剪贴；三是60句的设置，防止出错的自动处理。

（4）本程序仅写了40句评语，可以根据工作需要任意增减评语数目，相应的循环变量的终值也应作对应的改动。

（5）本程序虽然是一个草拟评语的程序，实际上可以以本程序为模式，改动 200 — 260 句中的信息内容，而完成其它工作的草拟工作，例如可以作为对工作人员的工作鉴定，人事介绍信，会议通知，文献摘要，甚至可以书写文章。这样，本程序设计思想可以为许多软件编制提供思路，只要存贮足够的信息，就可以大大减少汉字输入的工作量。

操作说明：

（1）程序运行后，打印评语和对应代号信息：

```
]RUN


1---尊敬师长
2---遵守校纪
3---学习认真
4---学习比较认真
5---爱动脑筋
6---工作积极
7---关心集体
8---团结同学
9---热爱劳动
10---能参加劳动
11---作业整洁
12---能完成作业
```

13---爱清洁
14---讲卫生
15---爱好文艺活动
16---上课发言积极
17---爱好体育活动
18---能帮助别人
19---敢于斗争
20---能参加学校活动
21---文明礼貌不够
22---不能遵守校纪
23---学习不认真
24---学习不够认真
25---不爱动脑筋
26---工作不主动
27---不太关心集体
28---不能团结同学
29---不热爱劳动
30---作业马虎
31---不能按时完成作业
32---不爱清洁
33---希望今后能够
34---尊敬师长
35---认真学习
36---团结同学
37---认真作业
38---帮助别人
39---在新学期争取进步
40---希望克服缺点好好学习

(2) 输入评语的方法,例如,老师对某学生作如下评语:尊敬师长,学习认真,爱动脑筋,关心集体,不能遵守

校纪，希望克服缺点，好好学习。

那么，只要查评语对应的代号，如

1——尊敬师长，

3——学习认真，

5——爱动脑筋，

7——关心集体，

22——不能遵守校纪，

40——希望克服缺点，好好学习。

这样，只要在340句的DATA语句中输入以下信息：

340 DATA 1,3,5,7,22,40,0

其中，DATA语句最后一个“0”标记，既表示对一个学生草拟评语的结束，也表示对另一个学生草拟评语的开始。

完整的程序清单（见H25）和6个学生品德评语的实例：

```
]LIST

2  REM H25
4  PRINT : PRINT
5  DIM A$(60)
10  REM 学生评语草拟程序
20  FOR I = 1 TO 40
30  READ A$(I)
35  PRINT I;"---";A$(I)
40  NEXT I
45  PRINT : PRINT
50  REM 输入评语<1,2,3,....40> 放入DATA中
60  ONERR  GOTO 110
70  FOR I = 1 TO 40
80  READ X
```

```
90  IF X = 0 THEN  PRINT : PRINT : PRINT : G
OTO 60
95  IF X =  - 1 THEN  END
110  PRINT A$(X);" ";
120  NEXT I
200  DATA 尊敬师长,遵守校纪,学习认真,学习比
较认真,爱动脑筋
205  DATA 工作积极,关心集体,团结同学,热爱劳
动,能参加劳动
210  DATA 作业整洁,能完成作业,爱清洁,讲卫生,
爱好文艺活动,上课发言积极
215  DATA 爱好体育活动,能帮助别人,敢于斗争,
能参加学校活动
230  DATA 文明礼貌不够,不能遵守校纪,学习不认
真,学习不够认真,不爱动脑筋,工作不主动
235  DATA 不太关心集体,不能团结同学,不热爱劳
动,作业马虎,不能按时完成作业,不爱清洁
250  DATA  希望今后能够,尊敬师长,认真学习,团
结同学,认真作业,帮助别人
260  DATA 在新学期争取进步,希望克服缺点好好
学习
300  DATA 2,5,7,12,38,0
310  DATA 1,3,28,40,0
320  DATA 2,4,18,33,37,0
330  DATA 4,7,21,31,36,0
340  DATA 1,3,5,7,22,40,0
350  DATA  6,8,11,19,27,-1
]RUN

遵守校纪 爱动脑筋 关心集体 能完成作业 帮助别
```

尊敬师长 学习认真 不能团结同学 希望克服缺点
好好学习

遵守校纪 学习比较认真 能帮助别人 希望今后能
够 认真作业

学习比较认真 关心集体 文明礼貌不够 不能按时
完成作业 团结同学

尊敬师长 学习认真 爱动脑筋 关心集体 不能遵守
校纪 希望克服缺点好好学习

工作积极 团结同学 作业整洁 敢于斗争 不太关心
集体

三、编 程 方 法

1. 流程框图方法

学习BASIC语言和其它高级语言(FORTRAN、ALG-OL、COBOL、PASCAL)一样，最终目的都是要编制解决实际问题的程序。“语言好学，程序难编”。这是一部分初学者经常谈及的问题。对于一个实际问题,常常感到无从下手，不知道怎样把一句句个别语句、函数、表达式组成一个完整的程序；不了解应如何设置变量，怎样安排数组，究竟用那种方式输入数据；甚至在选择语句时不知道前后语句如何安排，程序如何连接；……等等，诸如此类的问题。因而，使一部分同志感到程序设计变幻莫测，高不可攀，其实，这些顾虑是完全不必要的。

人们在处理问题的时候，常常事先安排好解决问题的步骤，同样，你要计算机来解题，也需要用计算机“听得懂的语言”来说明，要它干什么，不能干什么，都要一条一条顺序写清楚，只有这样，才能使计算机执行各种操作，有步骤地完成人们的各种意图。人们写不清楚的问题，计算机就无法执行并可能给出错误的结果。对实际问题缺少完整而周密的思索，拿起笔来就写，不讲思路，不讲技巧,不考虑算法，不注意语句的搭配，程序前后的连接，语法规则的正确使用和语句功能的了解,必然导致程序难于执行，结果事倍功半.这是编程困难或失败的根本原因。

程序设计有它自己特有的规律；编制程序是一个创造性劳动，只有熟悉程序设计的规律，努力学好语言，通过不断的实践，程序设计入门并不很难。

我们这里主要介绍的是程序设计的常用方法，即所谓程序流程图（FLOWCHART)。通过这个方法的介绍，会帮助你通过编程“关”,你将会知道，学会并习惯掌握流程图（又称框图）方法，是通过编程关的捷径之道。

人们常说，会画框图，就会编程序，这是颇有道理的。事实上，一个实际问题，画出它的框图，就等于完成程序设计的大部分任务，剩下来的事情，就是按照算法语言抄写框图的工作。

程序设计的正确性，在很大程度上取决于框图设计的质量。框图体现了程序设计的思想，反映了各个问题之间的相互联系，表达了各个子问题之间的作用，确定了解决问题的过程，以及程序的基本结构和走向等。因此，画了框图既形象又直观。它便于自己和别人阅读、理解、修改和检查程序。绘制框图，是各种高级语言程序设计中广泛使用的方法。

为了便于学习，我们只用两种编制框图的符号：矩形框和菱形框，菱形框为判断框符号，其它如读取数据、打印输出、变量赋值等等均用矩形框来表示。

现在，我们举几个简单的程序设计实例。

例1,比较两个数的大小。

设计考虑：设置两个变量A和B，它们之间的比较有三种情况：A＞B，A＜B，A＝B，即：

$$
\text{A 与 B 比较} \rightarrow A < B\,? \begin{cases} \text{YES} \rightarrow \text{打印} A < B \\ \text{NO} \rightarrow \begin{cases} A > B\,? \rightarrow \text{YES} \rightarrow \text{打印} A > B \\ \quad \downarrow \text{NO} \\ A = B \rightarrow \text{打印} A = B \end{cases} \end{cases}
$$

为了把上述思路形象地描述出来，让自己或别人看起来直观明了，叫人感到思路清楚，层次分明，来龙去脉，一目了然，我们就用几何图形和简单的文字说明来描述上述问题的计算过程，即框图的方法，见图3.1。

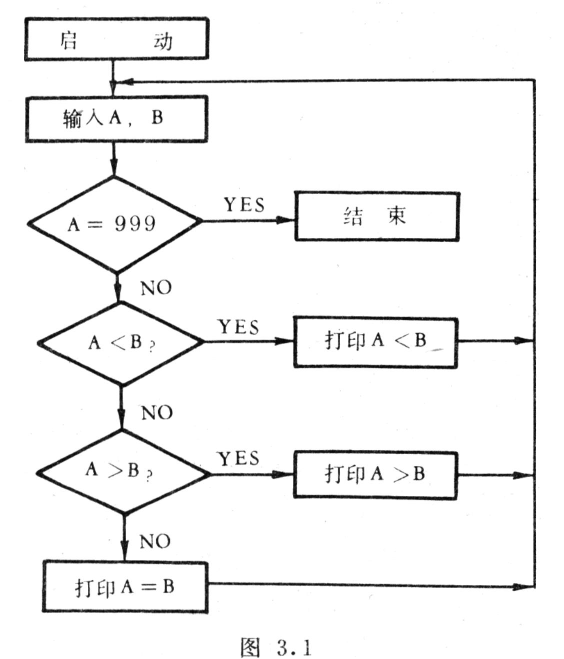


图 3.1

对照上述思路（它已经在框图中反映出来了），我们可以

看到A、B两个数的比较。

- 用INPUT语句来输入；
- 用终止标志来结束；
- 用条件语句来判断；
- 用GOTO语句来循环；
- 用打印语句来输出。

对照框图，我们不难写出程序，见H 26：

```
]LIST

20  REM H26
30  INPUT A,B
35  IF A = 999 THEN 130
40  IF A < B THEN 70
50  IF A > B THEN 90
60  PRINT A;"IS EQRAL TO";B
65  GOTO 30
70  PRINT A;"IS LESS THAN";B
80  GOTO 30
90  PRINT A;"IS GREATER THAN";B
100  GOTO 30
130  END
```

例2,已知5位学生的学号、BASIC语言和ENGLISH成绩，当平均分在0—59时给评语FAIL，60—79时给PASS，80—89时给GOOD，90—100给VERY GOOD(平均分数小数部份四舍五入)。给出框图，写出程序。

对这个问题，我们的主要思路如下：

(1) 应该将学号和成绩的信息送入计算机，这通常用三种提供数据语句的一种，本例选用READ语句，当然不要

忘记其搭配语句是DATA；

(2) 由于题意要求平均分数小数部份四舍五入，故用下述算式解决：X＝INT((A＋B)/2＋0.5)，实际上是选用的赋值语句；

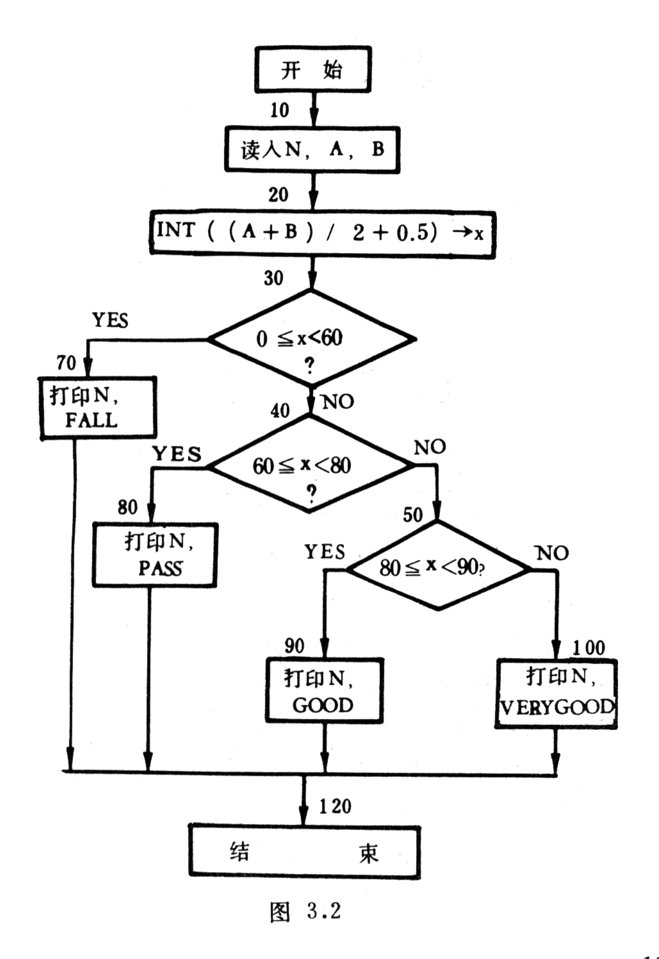


图 3.2

（3）为了给出不同平均成绩的评分。自然想到用条件语句IF/THEN来判断，例如当平均成绩X大于等于80而小于90时，给出“GOOD”的评语，而X等于或超过90时，则给出“VERY GOOD”的评语，其它各分数挡的评语仿此办理。

根据上述思路，不难画出主要框图，见图3.2：

对照框图，我们可以写出程序H27：

```
]LIST

2  REM H27
5  FOR I = 1 TO 5
10  READ N,A,B: PRINT N,A,B
20 X =  INT ((A + B) / 2 + 0.5)
25  PRINT "X=";X
30  IF X < 60 THEN 70
40  IF X < 80 THEN 80
50  IF X < 90 THEN 90
60  PRINT N;" ";"VERY GOOD": GOTO 100
70  PRINT N;" ";"FAIL": GOTO 100
80  PRINT N;" ";"PASS": GOTO 100
90  PRINT N;" ";"GOOD"
100  NEXT I
110  DATA 8501,88,86,8515,75,68,8504,98,92,8
521,60,58,8517,81,82
120  END


]RUN
8501            88              86
X=87
```

```
8501 GOOD
8515          75          68
X=72
8515 PASS
8504          98          92
X=95
8504 VERY GOOD
8521          60          58
X=59
8521 FAIL
8517          81          82
X=82
8517 GOOD
```

说明：

(1) 请注意框图近旁的行编号，编号标在每一个小框图旁有三个作用：一是对照框图编号及框图内的说明，可以方便地书写程序；二是有助于将程序框图与其相应的程序语言进行核对；三是若发生程序不能工作时，可以参照行编号检查其错误地点和性质并修正之。

(2) 当程序员确定了一张程序框图后，他就掌握了给计算机编写指令的整体方案，他可以通过假设的某些输入数据来检验他的方案的正确性，也可以在适当的地方加上某些主要结果的打印语句，这样，既可以及时发现错误，也可以了解程序的详细执行过程（例如，我们在上述程序H 27中添加了25句）。

(3) 编制框图的另一个目的，就是为以后的使用者提供了程序文本的资料，这会大大节省使用者理解程序的时间。读懂别人书写的程序不是一件轻而易举的事，而一个好的程

序框图，会给使用者带来很大方便。

对照程序画框图，或对照框图写程序，是初学者应该掌握的基本功。

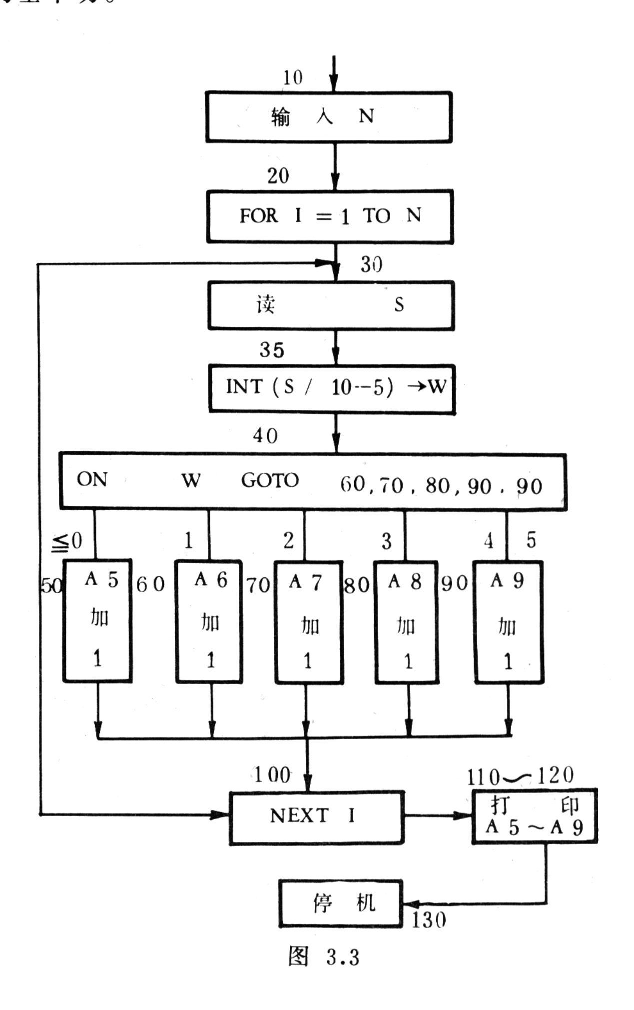


图 3.3

例 3，统计N个学生考试成绩，打印90分以上，80—89，70—79，60—69，60以下学生的人数。

设计思想：

安排一重循环，循环变量为I，取值范围为1到N，读成绩S，按W＝INT(S/10－5)算式处理，用开关语句ON…W控制，以区分不同分数档，学生人数放在A5～A9中

为此，画出框图，见图3.3。

相应程序H28：

```
]LIST

5  REM H28
10  INPUT "N=";N
20  FOR I = 1 TO N
30  READ S
35 W =  INT (S / 10 - 5): IF W < 0 THEN W =
0
40  ON W GOTO 60,70,80,90,90
50 A5 = A5 + 1: GOTO 100
60 A6 = A6 + 1: GOTO 100
70 A7 = A7 + 1: GOTO 100
80 A8 = A8 + 1: GOTO 100
90 A9 = A9 + 1: GOTO 100
100  NEXT I
110  PRINT "A5=";A5: PRINT "A6=";A6
120  PRINT "A7=";A7: PRINT "A8=";A8: PRINT "
```

```
A9=";A9
130  END
140  DATA 89,57,95,74,72,63,90,83,90,100,53,
64,75,88,89,79,93,69,25,74
```

```
]RUN
N=20
A5=3
A6=3
A7=5
A8=4
A9=5
```

注：上面的几个程序和相应的框图，在实际上有微小差别，这在编制框图时是允许的，但主要程序设计思想，数学模型，变量或数，安排时必须在程序框图中标明清楚。

例4,求任意数的阶乘(如4,7,9,3,11,8,10）主要设计思想：

程序可以分两大部分来写，主程序为读数、计数（一共7次),打印各数阶乘；子程序为完成求任意数的阶乘。

就求阶乘来说，它有通用程序，所以我们把它放在子程序中，对重复多次的计算工作可以用循环FOR/NEXT语句，也可以用子程序的方法。

框图也分两部分画,这样比较清晰,见图3.4和图3.5：

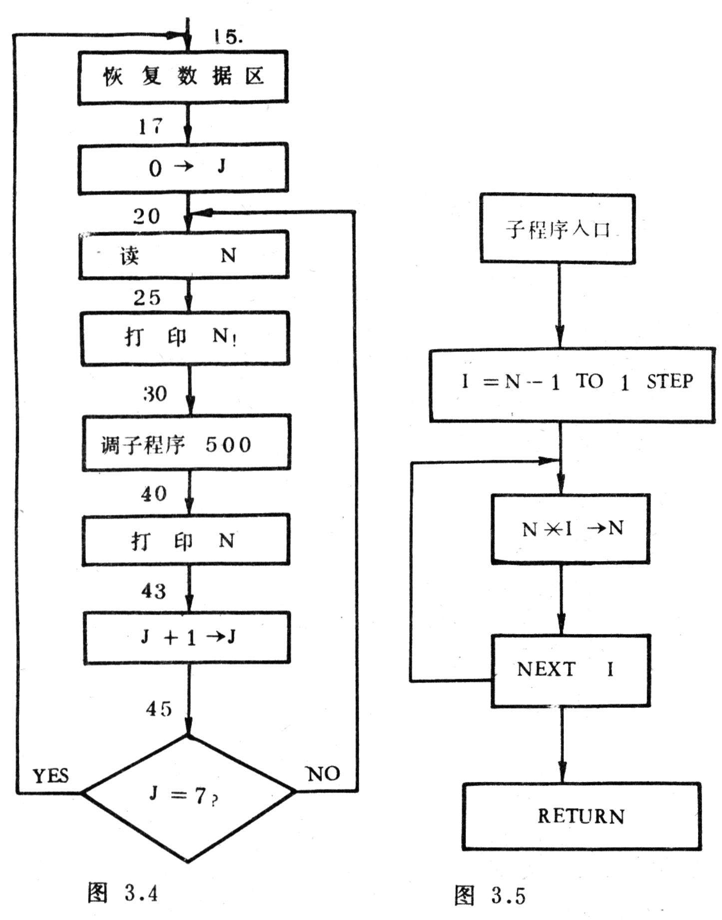


图 3.4　　　图 3.5

程序清单（见H 29）和运行结果为：

```
]LIST

5  REM H29
10  REM MAIN PROGRAM
```

```
20  READ N
25  PRINT N;"!=";
30  GOSUB 500
40  PRINT N: PRINT
43 J = J + 1
45  IF J = 7 THEN  END
50  GOTO 20
60  DATA 4,7,9,3,11,8,10
500  REM SUBROUTINE
510  FOR I = N - 1 TO 1 STEP  - 1
520 N = N * I
530  NEXT I
540  RETURN
    ]RUN
    4!=24

    7!=5040

    9!=362880

    3!=6

    11!=39916800

    8!=40320

    10!=3628800
```

2．模块化程序设计

模块化程序设计方法，是将一个大的程序划分成为若干功能相对独立的程序子模块。这种想法并不难理解，就好象一个程序总是由若干个完成不同功能的程序段组成，一个主

程序可以包括多个子程序一样。

作为程序的一个模块，应具有以下性质：

(1) 正确性：即使脱离该模块所在的程序环境，也能正确运行。

(2) 完整性：模块在逻辑上是完整的，其功能和作用是明确的。

(3) 通用性：模块作为一个程序段是通用的。

(4) 独立性：模块内部的操作和调整不影响程序的全局。

下面我们介绍一个模块化程序设计的实例，程序虽长一点（共有 100多句），但比较好读。介绍这段长程序有下列几个目的：

· 可以作为学过BASIC语言的读者复习之用，它几乎包括了基本BASIC语言的所有语句和功能运用。

· 对初学BASIC语言的读者，在学习了BASIC语言各语句及命令的功能与使用方法后，可以对照检查自己，实际掌握得怎么样，是否能看懂程序连接方法，来龙去脉。

· 对于学过语言，而丢的时间较长又缺少上机实习的同志，是否可以起到温故而知新的作用。

· 对于缺少编程练习，感到程序设计还比较陌生的同志，是否可以学到一些剖析程序的方法。

· 当然主要目的还在于，学习并掌握模块化程序设计的方法，借鉴其思路与技巧，从而得心应手地编制自己的程序。

为便于初学者比较顺利地阅读程序，我们除给出运行结果外，还对各程序子模块作了简要的说明，并附以粗略的框图。

完整的程序清单（见H30）和运行结果：

```
690  REM H30
700  CLEAR
705  DIM N(10),F$(10),M$(10),S(10),U(10),Y(1
0),Z(10),P(4,5),T(20)
710  REM  MAIN PROGRAM
720  INPUT "###";X
730  IF X < 0 THEN 760
740  ON X GOSUB 800,900,1000,1100,1200,1300,
1400,1500,1600,1700,1800,2100,2200
750  GOTO 720
760  END
```

700～760′

总控

输入功能号x，

根据x值（0～13）

转向子程序模块或

结束。

```
800  REM  SUB-(1)
820  FOR K = 1 TO 10
830 N(K) =  INT (20 + 10 *  RND (1))
840  PRINT N(K);" ";
850  NEXT K
855  PRINT
860  RETURN
```

800～860

产生10个两位随机整数，

并存贮在数组N（K）中。

```
900  REM SUB-(2)
910 F = 15
920  FOR I = 1 TO 10
930  IF N(I) <  = F THEN 950
940 F = N(I)
950  NEXT I
960  PRINT F
970  RETURN
```

900～970

将10个存贮在N（I）

中的最大值打印出来。

```
1000  REM SUB-(3)
1010 S = 0
1020  FOR J = 1 TO 10
1030 S = S + N(J)
1040  NEXT J
1050  PRINT S / 10
1060  RETURN
```

1000～1060

求存贮在N（I）中的10个数的平均值。

```
1100  REM  SUB-(4)
1120 M = 1
1130  FOR I = 1 TO 5
1140 M = M * I
1150  NEXT I
1160  PRINT M
1170  RETURN
```

1100～1170

求N(5)中的值的累乘积。即求M＝π N(5)。

```
1200  REM SUB-(5)
1210  FOR P = 1 TO 10
1220  IF  INT (N(P) / 3) = N(P) / 3 THEN 124
0
1230  GOTO 1250
1240  PRINT  SQR (N(P))
1250  NEXT P
1255  PRINT
1260  RETURN
```

1200～1260

求能被3整除的N(I)的值，并打印其开平方的值。

```
1300  REM SUB-(6)
1310  FOR M = 1 TO 10
1320  IF N(M) >  = 25 THEN 1340
1330  GOTO 1350
1340  PRINT N(M);" ";
1350  NEXT M
1360  PRINT
1370  RETURN
```

1300～1370

求数组N（M）中大于或等于25的值。

```
1400  REM SUB-(7)
1410  FOR I = 1 TO 9
1420  FOR J = I + 1 TO 10
1430  IF N(I) >  = N(J) THEN 1450
1440 Q = N(I):N(I) = N(J):N(J) = Q
1450  NEXT J
1460  NEXT I
1470  FOR I = 1 TO 10
1480  PRINT N(I);" ";
1490  NEXT I
1495  PRINT
1498  RETURN
```

1400～1498

将数组N(I)中的值按从大到小排列（共10个）。

```
1500  REM SUB-(8)
1510  FOR K = 10 TO 1 STEP  - 1
1520  PRINT "*";" ";K;"-";N(K)
1530  NEXT K
1535  PRINT
1540  FOR K = 1 TO 10 STEP 2
1550  PRINT N(K);" ";
1560  NEXT K
1570  PRINT
1580  RETURN
```

1500～1580

打印数组N（K）中的10个值，从N（10）到N(1)。

打印数组N（K）中K＝1,3,5,7,9的值。

```
1600  REM SUB-(9)
1620  FOR I = 1 TO 5
1630  INPUT "F$(I)=";F$(I)
1635  PRINT F$(I)
1640  NEXT I
1650  PRINT
1670  RETURN
```

1600～1670

键盘输入 5 组字符，输入一组打印一组。

```
1700  REM SUB-(10)
1710  INPUT "F$=?";F$
1720  IF F$ = "NO" THEN 1770
1730  FOR I = 1 TO 10
1740  IF F$(I) = F$ THEN  PRINT F$(I): GOTO
1770
1750  NEXT I
1760  PRINT "NOT FIND"
1765  GOTO 1710
1770  RETURN
```

1700～1770

查找存贮在字符串变量 F $ (I)中的字符，若没有打印出找不到信息。

1800～1830

输入10个货物的代号和数量，并打印出全部信息

```
1800  REM SUB-(11)
1810  FOR I = 1 TO 10
1820  INPUT M$(I),S(I)
1825  PRINT M$(I);" ";S(I)
1830  NEXT I: PRINT
```

```
1860  FOR I = 1 TO 9
1870  FOR J = I + 1 TO 10
1880  IF S(I) >  = S(J) THEN 1910
1890 T = S(I):S(I) = S(J):S(J) = T
1900 D$ = M$(I):M$(I) = M$(J):M$(J) = D$
1910  NEXT J
1920  PRINT M$(I);" ";S(I)
1930  NEXT I
1940  PRINT M$(10);" ";S(10): PRINT
```

1860～1940

将10种货物按数量多少从多至少排序，用的是枚举分类法。

```
1950  INPUT "P=";P
1955  IF P =  - 5 THEN 2055
1960 I = 0:J = 11
1970  IF J - I <  = 1 THEN 2040
1980 K =  INT ((J + I) / 2)
1990  IF P = S(K) THEN 2020
2000  IF P > S(K) THEN J = K: GOTO 1970
2010 I = K: GOTO 1970
2020  PRINT M$(K);" ";P
2030  GOTO 1950
2040  PRINT P;" ";"NOT FIND"
2050  GOTO 1950
2055  PRINT "-*-*-*-*-*-"
2060  RETURN
```

1950～2060

查找前面存贮的货物情况，按数值查找，用的是对分搜索法，这种方法查找速度比线性搜索快的多。

```
2100  REM SUB-(12)
2105 S = 0
2110  FOR I = 1 TO 10
2120  INPUT "U(I)=";U(I): INPUT "Y(I)=";Y(I)
2130 Z(I) = (U(I) * U(I)) - (Y(I) * Y(I))
2140 S = S + Z(I)
2150  NEXT I
2160  PRINT S
2170 D =  SQR ( ABS (S))
2180  PRINT "D=";D
2190  RETURN
```

2100～2190

输入10组数，每组2个值，求两数平方差的和，并打印其开方值。

```
2200  REM SUB-(13)
2210  FOR I = 1 TO 4
2215  FOR J = 1 TO 5
2220  INPUT "P(I,J)=";P(I,J)
2225  NEXT J
2230  NEXT I
2232  PRINT
2235  FOR I = 1 TO 4
2240  FOR J = 1 TO 5
2245  PRINT P(I,J);" ";
2250  NEXT J
2255  PRINT
2257  NEXT I
2259  PRINT
2260  FOR I = 1 TO 4
2270 T(I) = 0
2280  FOR J = 1 TO 5
2290 T(I) = T(I) + P(I,J)
2300  NEXT J
```

2200～2350

本段模块完成三个功能：

① 输入四行五列的值并打印一个二组数表；

② 求每一行之素值的累加和；

③ 求整个两组数表所有之素值的和。

```
2310  PRINT I;" ";"T(";I;")=";T(I)
2312  PRINT
2320  NEXT I
2330 S = 0
2335  FOR I = 1 TO 5
2340 S = S + T(I)
2345  NEXT I
2348  PRINT "S=";S
2350  RETURN

]RUN
###1
26 27 24 22 22 29 25 22 25 21
###2
29
###3
24.3
###4
120
###5
5.19615243
4.89897949
4.5825757

###6
26 27 29 25 25
###7
29 27 26 25 25 24 22 22 22 21
###8
* 10-21
* 9-22
* 8-22
```

```
* 7-22
* 6-24
* 5-25
* 4-25
* 3-26
* 2-27
* 1-29

29 26 25 22 22
###9
F$(I)=BASIC
BASIC
F$(I)=RETURI
RETURN
F$(I)=GOSUB
GOSUB
F$(I)=BSAVE
BSAVE
F$(I)=CATALOG
CATALOG
```

```
###10
F$=?BSAVE
BSAVE
###10
F$=?LOCK
NOT FIND
F$=?CATALOG
CATALOG
###11
?A1,98
A1 98
?A2,88
A2 88
?A3,79
A3 79
?A4,55
A4 55
?A5,49
A5 49
?A6,77
A6 77
?A7,81
A7 81
?A2,90
A2 90
?A9,60
A9 60
?A10,55
A10 55
 A1 98
 A2 90
```

```
A2 88
A7 81
A3 79
A6 77
A9 60
A4 55
A10 55
A5 49
P=79
A3 79
P=49
A5 49
P=100
100 NOT FIND
P=-5
-*-*-*-*-*-
###12
U(I)=10
Y(I)=6
U(I)=20
Y(I)=5
U(I)=11
Y(I)=6
U(I)=20
Y(I)=15
U(I)=33
Y(I)=23
U(I)=50
Y(I)=10
U(I)=1
Y(I)=2
```

```
U(I)=3
Y(I)=4
U(I)=5
Y(I)=6
U(I)=7
Y(I)=8
3623
D=60.1913615
###-3

]RUN
###13
P(I,J)=12
P(I,J)=23
P(I,J)=34
P(I,J)=45
P(I,J)=56
P(I,J)=21
P(I,J)=32
P(I,J)=43
P(I,J)=54
P(I,J)=65
P(I,J)=10
P(I,J)=20
P(I,J)=30
P(I,J)=40
P(I,J)=50
P(I,J)=60
P(I,J)=70
P(I,J)=80
P(I,J)=90
```

```
P(I,J)=11

12 23 34 45 56
21 32 43 54 65
10 20 30 40 50
60 70 80 90 11

1 T(1)=170

2 T(2)=215

3 T(3)=150

4 T(4)=311

S=846
###-3
```

关于程序H30有以下几点说明：

(1) 由于整个程序采用模块化程序结构，数组不能重复说明，而是开的不同的存贮，如N(10)，F$(10)，M$(10)，U(10)，P(4,5)等等。

(2) 每一个子程序的最后一句，亦即每个子模块的最后一句，一定不要忘记RETURN语句。

(3) 正如前面所述的，本程序作为一种复习，是为了练习阅读程序的能力，故各模块之间已经不受独立性的约束，各子程序之间可以交叉，例如SUB—(2)的程序，用到SUB—(1)中的存贮结果。

(4) 从模块的独立性来看，你可以单独调出某一模块的程序自由运行，而不会影响程序的全局，例如敲入RUN 2200，可以单独打出二组数表，各行元素总和，以及方阵全体元素的总值（这里采用INPUT P(I,J)输入数据）。在

上述工作过程后，由于要执行2350 RETURN语句，而发生机器出错，这也无关大局，按停止键，即可重新工作。

3. 实用数据处理原理程序

模块化程序结构是一种先进的程序设计方法。它的主要思想是将一个大的程序划分为若干功能模块，并供用户选择，但是究竟选用哪一个功能，完全由操作者自定。因此，设计一个选择程序，供操作者使用，就显得十分必要。方法是在程序一开始，计算机自动显示各功能名称，用户根据需要键入希望完成的功能号，则程序立即转入相应的模块，由于这种方法酷似餐馆中的点菜方式，故称“菜单”技术。

程序H31是一个实例，它采用多择一结构。

```
]LIST

5  REM H31
10  REM 数据处理原理
20  DIM D(20),A(9),B(11),X(19)
30  RESTORE
40  PRINT "1. 数据插入"
50  PRINT "2. 数据删除"
60  PRINT "3. 数据修改"
70  PRINT "4. 数据合并"
80  INPUT K$
90  IF K$ = "#" THEN  END
100 MO =  VAL (K$)
110  IF MO < 1 OR MO > 4 THEN 80
120  ON MO GOSUB 140,370,560,750
130  HOME : GOTO 30
```

程序一经运行，立即显示各功能项:

```
]RUN
1. 数据插入
2. 数据删除
3. 数据修改
4. 数据合并
?
```

这样，操作者或者不是编制本程序的使用者，就可以了解到本程序有四个功能，它可以完成诸如数据插入、删除、修改、合并的操作。

80句中的INPUT K$，是等待用户响应，由操作者输入希望完成的功能代号，在1到4之间。

110句是防止程序乱跑(BOMB PROOFING)。上述4种功能是用1到4这四个数字来响应的，若使用者可能按错数字键，则程序可能会“爆炸”,所以安排了“防爆”措施。这样做，实际上是增加了是否在1到4以内的逻辑检查，如出错，则自动返回重新输入。它虽然不是文件管理上提出的要求，但它是属于提高程序质量的方法。

程序中的输入为什么采用字符串变量K$，而不用简单变量呢？这基于两方面的考虑：其一是防止和程序中处理的数值相混淆；其二是输入方式灵活方便，既可以输入字符，也可以输入数值。

例如，要结束本程序可以输入“#”号，经90句处理结束停机。而输入的“1”，“2”，“3”，“4”,也可以理解是字符，经100句VAL(K$)函数处理变成数值。

120句是程序在分析用户的要求后,所作的分支处理，显然用开关控制语句ON/GOSUB最为合适。而120句后面

的行号，是程序H31中相应 4 个功能模块的人口地址。

每个独立的程序模块，分别完成注释中的相应功能，而在每个模块结尾处都设置了 RETURN语句，以便在完成本模块功能后，正确返回总控，即经130句先清屏，再转向“菜单”提示，从而根据需要再作其它操作。

下面给出一个实用数据处理的原理程序（H32)，读者可以从中体会“菜单”技术在实际编程中的应用，模块化的程序结构以及数据处理几种基本操作。关于每一操作的原理说明，请读者自行分析。

```
]LIST

5  REM H32
10  REM 数据处理原理
20  DIM D(20),A(9),B(11),X(19)
30  RESTORE
40  PRINT "1. 数据插入"
50  PRINT "2. 数据删除"
60  PRINT "3. 数据修改"
70  PRINT "4. 数据合并"
80  INPUT K$
90  IF K$ = "#" THEN  END
100 MO =  VAL (K$)
110  IF MO < 1 OR MO > 4 THEN 80
120  ON MO GOSUB 140,370,560,750
130  HOME : GOTO 30
140  REM 数据插入
150  FOR I = 1 TO 10
160  READ D(I)
170  PRINT D(I);" ";
180  NEXT I
```

```
190  PRINT
200 K = 0
210  INPUT "N=";N: REM 数据插入位置
220  IF N =  - 1 THEN 360
230  FOR I = 10 + K TO N STEP  - 1
240 D(I + 1) = D(I)
250  NEXT I
260 K = K + 1
270  INPUT "A=";A: REM 插入的数据
280 D(N) = A
290  PRINT
300  FOR I = 1 TO 10 + K
310  PRINT D(I);" ";
320  NEXT I
330  PRINT
340  GOTO 210
350  DATA 12,47,35,26,61,83,90,55,27,88
360  RETURN
370  REM 数据删除
380  FOR I = 1 TO 10
390  READ D(I)
400  PRINT D(I);" ";
410  NEXT I
420  PRINT
430 K = 0
440  INPUT "N=";N: REM 删除的位置
450  IF N =  - 1 THEN  RETURN
460  FOR I = N TO 9
470 D(I) = D(I + 1)
480  NEXT I
490 K = K + 1
```

```
500  PRINT
510  FOR I = 1 TO 10 - K
520  PRINT D(I);" ";
530  NEXT I
540  PRINT
550  GOTO 440
560  REM 数据修改
570  FOR I = 1 TO 10
580  READ D(I)
590  PRINT D(I);" ";
600  NEXT I
610  PRINT
620  REM 870-920连续修改
630  FOR I = 1 TO 10
640  INPUT "N=";N: REM 修改的位置
650  IF N =  - 1 THEN 690: REM 修改结束，显
示，返回
660  INPUT "M=";M: REM  修改的数据
670 D(N) = M
680  NEXT I
690  PRINT
700  FOR I = 1 TO 10
710  PRINT D(I);" ";
720  NEXT I
730  FOR I = 1 TO 2000: NEXT I
740  RETURN
750  REM 数据合并
760  IF MO = 4 THEN  FOR I = 1 TO 10: READ S
: NEXT I
770  FOR I = 1 TO 8: READ A(I): NEXT
780  FOR I = 1 TO 10: READ B(I): NEXT
```

```
790 I = 1:J = 1:N = 0
800 N = N + 1
810  IF A(I) > B(J) THEN  LET X(N) = A(I):I
= I + 1: GOTO 830
820 X(N) = B(J):J = J + 1
830  IF N < 18 THEN 800
840  FOR I = 1 TO 18
850  PRINT X(I);",";
860  NEXT
870  PRINT
880  DATA 98,97,89,85,84,71,69,63
890  DATA 96,94,92,90,87,83,72,69,60,55
900  FOR I = 1 TO 2000: NEXT I
910  RETURN
```

程序结构的几点说明：

（1）本程序分为五段，各段功能如下：

· 10—130句：菜单提示，总控，防止出错处理，数组安排。

· 140—360句：数据插入。

· 370—550句：数据删除。

· 560—740句：数据修改。

· 750—910句：数据合并。

（2）本程序各功能块所用数组，全部放在程序的第一段20句，这样安排的目的是，防止多次运行某一功能块时，出现数组重复说明而中止程序运行的错误。例如，若在程序第四段安排一个565句，DIM D(20),则重复运行第四段程序时发生数组说明的错误，而第四段作为一个独立的程序编制时，则565DIM D(20)是不能省略的。

（3）程序第一段30句安排了恢复数据区语句RESTORE，这是整个五段程序连接在一起时必不可少的，否则各分模块完成该模块功能后，再恢复总控时，其它各模块无数可读。

（4）130句设置了一个清屏语句HOME，这是为了屏幕显示干净，否则菜单提示和运行结果混在一起，有些信息丢失。但在打印程序结果时，应删去HOME语句，而改为130 PRINT：PRINT：GOTO 30，否则发生打印格式不统一以及打印机空走等情况，中华学习机就属于此例。

（5）730句和900句，安排了一个延时技术，即用FOR I=1 TO 2000：NEXT I 的空循环，目的是看清运行结果，否则由于迅速返回总控显示菜单而看不清运行结果。这是130句HOME指令造成的。

（6）760句安排了一个虚读技术，这是因为第五段程序是处理两组数据的合并问题，这两种数据放在880和890两个DATA语句中，如果不安排760句的条件判断和虚读技术，则发生350句的数据和880句的数据合并，从而造成程序错误并中断。

下面给出本程序H32运行的部份实例，读者不难根据程序中的中文注释，正确执行操作。

```
]RUN
1. 数据插入
2. 数据删除
3. 数据修改
4. 数据合并
?1
12 47 35 26 61 83 90 55 27 88
```

```
N=1
A=444

444 12 47 35 26 61 83 90 55 27 88
N=5
A=777

444 12 47 35 777 26 61 83 90 55 27 88
N=-1
```

程序运行后，首先出现菜单和?,从键盘键入 1，则程序转入数据插入功能模块，并打印原数据12，47，……，27，88的信息，若要在上述数据的第一个位置,插入444的数据，则分别键入 1 和444,则屏幕显示444,12，47，……，27,88,说明 444已经插入第一个位置中，而原数据信息自动向后移动。若继续插入新的信息，操作同上，如不再增加信息键入－1，程序自动返回，并重出现菜单和？。

```
1. 数据插入
2. 数据删除
3. 数据修改
4. 数据合并
?2
12 47 35 26 61 83 90 55 27 88
N=5

12 47 35 26 83 90 55 27 88
N=3

12 47 26 83 90 55 27 88
N=-1
```

键入 2 后，程序进入数据删除模块，显示原数据信息。输入 5，则原数据第 5 号位置上的61删去；输入 3，则又删

去35的数据。输入－1结束本功能模块操作，再返回总控，又出现菜单。

```
1. 数据插入
2. 数据删除
3. 数据修改
4. 数据合并
?3
12 47 35 26 61 83 90 55 27 88
N=1
M=333
N=2
M=777
N=-1

333 777 35 26 61 83 90 55 27 88
```

键入 3，进入数据修改模块，显示原数据信息，上面是在 1 号位置改成333，2 号位置改成 777 的结果，键入－1后显示修改后数据情况，再返回总控。

```
1. 数据插入
2. 数据删除
3. 数据修改
4. 数据合并
?4
98,97,96,94,92,90,89,87,85,84,83,72,71,69,69
,63,60,55,
```

键入 4，进入数据合并模块，立即显示原数据区中两组数据合并情况，并且按有序排列。

若结束整个程序键入“＃”号即可。

上述操作不一定按1,2,3，4 顺序执行，可以任意进行任

一功能的操作，每种操作也可以进行多次，只要不要超过数据存贮情况，从而使程序执行具有更大的灵活性，方便了用户。

4. 多重循环程序设计技巧

众所周知，由于计算机具有快速准确及逻辑判断的特点，几乎所有的高级程序设计语言都设置了循环语句，用以有效地解决大量的在处理形式又完全相同的重复性计算问题。对多重循环来说，这种重复计算的工作量是十分巨大的，多重循环最内层循环体的执行次数，一般来说是各层循环次数连乘积之值，由于这个数字往往非常庞大，所以对多重循环来说，十分注意减少不必要的循环次数，尽一切可能避免毫无意义的重复工作，特别是如有可能尽量减少循环嵌套的层次，就是一个有实际意义的工作。这样，我们就可能使计算机执行较少的命令，干出更多的工作，从而达到节约时间，少占存贮、加快速度、提高效率的目的。

对于嵌套的循环来说，外层循环的控制变量取一定数值后，内层循环则要整整转上一圈，只要内层循环不结束，就谈不上外层循环控制变量的增值和判断。这种循环语句的执行过程，和时钟的转动规律有十分相似的地方，以三重循环为例，三层循环的关系好像手表的时针、分针与秒针，时针移动一格，分针转动一圈，分针移动一格，秒针走动一周。但二者也有差异，时钟上足发条，指针的移动都是做的有用的工作，它在一秒一秒地计时；而程序的循环呢？并非都是执行有效的计算，事实上，常常发生不可能出现我们所希望的结果时，计算机还在继续进行毫无意义的循环。

这里从一个具体实例出发，谈谈多重循环的设计技巧。

用 100 元买 100 只鸡，已知每只小鸡、公鸡、母鸡分别为 0.5 元、 2 元、 3 元，问每只鸡至少买一只一共有多少种买法。

解：设 A、 B 、 C 分别为购买小鸡、公鸡和母鸡的只数，据题意有以下方程：

$$A + B + C = 100 \quad (1)$$

$$0.5A + 2B + 3C = 100 \quad (2)$$

而约束条件为：$1 \leqslant A、B、C \leqslant 98$

我们可以用循环的方法，取 A、 B 、C 的各种可能取值，去投试上述方程，只有全部满足方程(1)和(2)的 A、 B 、 C 值，才是问题的解，故容易写出程序 H33。

```
]LIST

10  REM H33
20  FOR A = 1 TO 100
30  FOR B = 1 TO 100
40  FOR C = 1 TO 100
50  IF A + B + C < > 100 THEN 80
60  IF A + 4 * B + 6 * C < > 200 THEN 80
70  PRINT A;B;C
80  NEXT C
90  NEXT B
100  NEXT A
110  END
```

上述程序总的循环次数高达100万次（$100 * 100 * 100 = 10^6$），所以运行时间是很长的，在 PC-1500机上要运行13个多小时，在 APPLE-Ⅱ机上要运行 2 个多小时，而在 WA-

NGPCS-Ⅱ和WANG2200T（均为美国计算机），运行了4个多小时。

程序H33并没有原则性错误，因为它确实可以得到正确的结果，为了更快地得到答案，显然还有值得我们探讨的问题。

(1) 第一个问题是A、B、C的终值是否都是100？

程序(1)中的三个循环变量A、B、C的终值均为100，是没有考虑每只鸡至少买一只这个约束条件编写的。事实上，各种鸡至少各买一只，每种鸡最多只能买98只。所以三个循环变量的终值都只要取98即可。对于每一个单循环来说，终值仅仅减小了2，但对多重循环来说，它的意义就非同小可。因为各重循环终值按98计算，总的循环次数变为$98\times98\times98=941192$次，比程序(1)循环次数减少58808次。也就是说，减少了近6万次的无效循环。

(2) 改写一下循环变量的终值

循环变量的初值、终值和步长可以是具体的数值，也可以是已赋值的变量，或者是表达式。从这个基本约定出发，我们把三重循环的终值改写一下：

```
20 FOR A = 1 TO 98
30 FOR B = 1 TO 99 - A
40 FOR C = 1 TO 100 - A - B
```

显然，这个程序段，和三重循环变量都从1变化到98，是等价的（注意：本程序段写法比较特殊，它是考虑到A、B、C三个变量均应满足$A+B+C=100$和$1\leqslant A、B、C\leqslant98$这两个条件的）。但在执行的效果上，却有很大的不同。

为了说明这个问题，我们先考察一下下面两个程序H34

及H 35。

```
]LIST

5  REM H34
10 N = 0
20  FOR A = 1 TO 5
30  FOR B = 1 TO 5
40  FOR C = 1 TO 5
50 N = N + 1
60  NEXT C
70  NEXT B
80  NEXT A
90  PRINT "N=";N
```

```
]LIST

5  REM H35
10 N = 0
20  FOR A = 1 TO 5
30  FOR B = 1 TO 6 - A
40  FOR C = 1 TO 7 - A - B
50 N = N + 1
60  NEXT C
70  NEXT B
80  NEXT A
90  PRINT "N=";N
```

上述两个程序假设都满足：

$A + B + C = 7$

$1 \leq A、B、C \leq 5$

这两个条件。

由于多重循环最内层循环体的执行次数，是各层循环次数连乘积之值，所以程序H 34总共执行循环 5 * 5 * 5 = 125次,程序的90行是统计打印循环总次数的。

对于程序H 35，根据多重循环的执行过程，不难计算出总的循环次教只有35次，它仅为程序H 34循环次数的28%。

由此可知，对本程序来说，循环终值的写法是大有讲究的。

现在我们再回到本题上来，改用程序H 36求解本题：

```
]LIST

10  REM H36
20  FOR A = 1 TO 98
```

```
30  FOR B = 1 TO 99 - A
40  FOR C = 1 TO 100 - A - B
50  IF A + B + C = 100 AND A + 4 * B + 6 * C
 = 200 THEN  PRINT A;B;C
80  NEXT C: NEXT B: NEXT A
```

程序H 36比程序H 33写得更为简捷，但最主要的是由于把循环终值形式上改写了一下，总循环次数已由程序H 33的100万次,改变到只有161700次，它仅为程序H 33循环次数的16.17%，显然，这是一个大的飞跃，它省去了838300次无用循环。

从上述事实,可以看到多重循环的工作量是非常巨大的，而缩短初值和终值之间的差距是减少循环次数的一个有效方法。当然，这种减少，必须由题意分析确定,不能任意减少。不同的题目，可以用增大初值、减少终值或两者兼而有之，来达到减少循环次数的目的，上例是用减少终值的方法来实现的。至于本例改写终值的形式，是受本题不定方程求解这个特殊条件制约的。

(3) 三个循环的终值也不可能都是98。

这个问题也是非常明显的，若买98只小鸡则共花去98×0.5＝49元，再买母鸡、公鸡各一只，又分别花去 3 元和 2 元，虽然买了 100 只鸡，但只花了49＋ 3 ＋ 2 ＝54元，不满足“百钱买百鸡”的题意，同理母鸡、公鸡也不可能买到98只，否则100元钱不够花。

让我们进一步思考，看看还有什么方法，来减少循环次数。

改写方程(1)和(2)：

$$A+B=100-C \tag{3}$$

$$A+4B=200-6C \tag{4}$$

(3)式×6－(4)式得：

$$5A+2B=400 \tag{5}$$

(4)式－(3)式得：

$$5C+3B=100 \tag{6}$$

(3)式×4－(4)式得：

$$3A-2C=200 \tag{7}$$

改写(5)、(6)、(7)三式则有：

$$\begin{cases} B=\dfrac{400-5A}{2} & (8) \\ C=\dfrac{100-3B}{5} & (9) \\ A=\dfrac{200+2C}{3} & (10) \end{cases}$$

表面上原来两个方程(1)、(2)，变成了三个方程(8)、(9)、(10)，多了一个方程，但经过这样处理，却使问题简单多了，因为每个方程只是一个变量依赖于另一个变量的单一关系。

由(8)式，B最少取1时（根据题意）：

$A=79.6$　　　　取整数79。

由(10)式，C最少取1时：

$A=67.3$　　　　取整数67。

故可知A的变化范围：$67\leq A\leq 79$。

再考察方程(2)，当A为奇数时，不满足方程的解，所以A必为偶数，这样，可知A的取值范围是68到78，且步长为2。

这个事实告诉我们，A 的终值不是98，A 的初值也不是1，并且A 的变化也是有规律可寻的，它的间隔是 2 。

读者不难想到，B 和C 的初、终值又是什么？它们是否也按一定的规律变化呢？这个问题，还是让计算机帮助我们判断吧！

（4）计算机可以帮助找到变量的取值范围和变化规律

A 的变化范围确定了，代入（8）式，则可确定B 的变化范围；同理B 的数值确定了，代入（9）式，也可确定C 的变化范围。

现在，我们不要一一投试方程去计算，只要编两个小程序H 37和H 38，让计算机自动去求解吧！

```
]LIST

5  REM H37
7  PRINT "B=";
10  FOR A = 68 TO 78 STEP 2
20 B = (400 - 5 * A) / 2
30  PRINT B;",";
40  NEXT A


]RUN
B=30,25,20,15,10,5,
```

由此可知，B 的变化范围是从 5 到30，且步长为 5 。

```
]LIST

5  REM H38
7  PRINT "C=";
10  FOR B = 5 TO 30 STEP 5
```

```
20 C = (100 - 3 * B) / 5
30  PRINT C;",";
40  NEXT B
```

```
]RUN
C=17,14,11,8,5,2,
```

程序结果表明，C的取值范围是从2到17而步长为3。

综合A、B、C的取值范围和步长变化规律，我们可以写出程序H39：

```
]LIST

10  REM H39
20  FOR A = 68 TO 78 STEP 2
30  FOR B = 5 TO 30 STEP 5
40  FOR C = 2 TO 17 STEP 3
50  IF A + B + C = 100 AND A + 4 * B + 6 * C
 = 200 THEN  PRINT A,B,C
60  NEXT C: NEXT B: NEXT A
```

读者不难发现，程序H39总共循环次数只有125次，仅为程序H33循环次数的万分之一点二五。显然，本程序的运行速度是很快的，在PC-1500机上仅运行20秒就得出全部结果。

由此可知，同时增大初值和缩小终值，并且加大步长值，会使程序效率大大提高，它们是加快程序循环部分执行速度的又一个好办法。

同时，我们还应看到，如果没有计算机的“帮助”，很快

地作出上述三个变量的取值范围及变化规律的判断，还是比较困难的。在程序设计中，利用计算机协助人们编程，也是一个值得研究的课题。

（5）循环嵌套的重数可以减少

减少循环的层次，使循环嵌套愈少愈好，这样，便可以进一步提高程序的运行速度。

分析程序H 39，我们用C = 100 − A − B 来代替40句，显然是可以的，因为当A 和B 的值确定后，C 的值也就确定下来，故可以省去一重循环，这样总的循环次数只有25次了。同样，我们从方程(8)、(9)、(10)中还看到这样一个事实，每个变量只依赖一个变量，A定B 也就定了，而B定C 也定了，因而我们完全可以再省去一重循环，即删去30行的一重循环，而改用B =（400 − 5 × A）/2的赋值语句。省去二重循环后，总共循环次数只有5 次，它仅为程序H 33循环次数的百万分之5，可见运行时间会进一步缩短，而速度则会大大提高。见程序H 40：

```
]LIST

10  REM H40
20  FOR A = 68 TO 78 STEP 2
30 B = (400 - 5 * A) / 2
40 C = 100 - A - B
50  IF A + B + C = 100 AND A + 4 * B + 6 * C
.= 200 THEN  PRINT A,B,C
60  NEXT A
```

（6）循环体的判断语句可以省去

经过上面好几步的简化，我们得到程序H 40，好像该做

的我们都做了，循环初值增大了，终值减小了，步长增加了，循环层次又减少了，作为循环程序的设计技巧，基本上我们都考虑到了，其实，就本题来说，循环体中的内容还可以斟酌。由于A、B、C之间关系完全由方程(8)、(9)、(10)所制约，并且在处理上述工作时，又已经考虑了约束条件，所以，程序H40中循环体内的判断语句完全可以省。因而有程序H41：

```
]LIST

10  REM H41
20  FOR A = 68 TO 78 STEP 2:B = (400 - 5 * A
) / 2:C = 100 - A - B: PRINT A,B,C: NEXT A


]RUN
68              30              2
70              25              5
72              20              8
74              15              11
76              10              14
78              5               17
```

这个程序不仅简短节省内存，而且运行速度又是所有程序中最快的一个。在PC-1500机上只运行了几秒钟，而在APPLE-Ⅱ机上执行时间仅为1秒钟左右。

综上所述，我们围绕一个不定方程的求解过程，探讨了循环程序设计技巧的各个方面。虽然本题是从一个具体例子出发，任意引出的结论却具有普遍的意义。循环是BASIC语言中的精华和核心，编程序几乎离不开循环，因此，掌握

循环程序的设计技巧，无论对学习哪一种高级语言都是重要的。循环程序的设计技巧可以归纳以下各个方面：在步长不变的情况下，减小初值和终值的差距是减少循环次数的有效方法；在初、终值一定的情况下，加大步长可以大大提高程序的效率；减少循环嵌套的重数，是提高程序运行速度的关键；调正和精简循环体的语句，是提高程序质量的有力措施；合理安排控制和转向语句，是及时脱离无效循环避免重复计算的前提。而所有这一切，又以尽量减少无用循环次数，努力避免大量无意义的重复工作为根本，这是高质量进行循环程序设计的关键。

四、打 印 输 出

1. 中英文状态转换

中华学习机在中文状态下打印输出时，其速度是比较慢的，无论是输出汉字还是输出英文字符，打印机都要来回打印两次。在英文状态打印输出时，其打印速度比较快，但所有中文汉字都变成了一些英文字母和莫名其妙的字符。

如何提高在中文状态下的打印速度，又使得中、西文字符清晰可辨，就是一个有实用意义的问题。

可以设想，如果设法使机器在打印汉字时按中文状态工作，而在打印其它字符时按英文状态工作，显然可以提高打印输出的速度。这就是说要对打印机实行某种控制，使其按上述要求自动进行中英文状态的转换。

键盘上中西文的转换，是通过改变“中文”，“英文”的两个按键，用手动的方法完成的，但这种操作不适用于对打印机的控制。因此，容易想到，要使打印机能自动进行中英文的转换，必须从程序上来进行控制。

如何在程序中进行中、英文打印输出的分别处理呢？

请先看以下一个实验。

如果屏幕上显示的是英文字幕，若此时键入PR#3命令并回车，则屏幕上立即显示中文状态的图象。这就给我们一个启发，在程序中利用DOS命令和中华学习机汉字系统提供的软开关，在程序的适当地方，按打印汉字还是英文

的需要，进行中文打印状态和英文打印状态的转换。

例如，要进入中文状态，可用程序H 42实现。

```
]LIST

 10  REM H42
 20 D$ =  CHR$ (4)
 30  PRINT D$;"PR#3"
 40  POKE 1659,1
 50  PRINT "中文信息"
```

而进入英文状态，可连接以下语句：

```
]LIST

70  PRINT D$;"PR#1"
80  PRINT "英文信息"
```

程序H 43是一个中英文状态转换程序控制的实例：

```
]LIST

5  REM H43
10  REM 中英文状态转换
20 D$ =  CHR$ (4)
30  POKE 1659,5
40  PRINT "中华学习机在中文状态下打印速度
较慢，若要提高打印速度，可以设法使机器在打
印汉字时按中文状态工作，而打印其它字符时按
英文状态工作."
50  PRINT "为此，必须使程序按需要进行两种
状态的转换."
60  PRINT "下面是一个实例."
70  POKE 1659,0
```

```
80  PRINT
90  PRINT D$"PR#3": REM 进入中文系统
100  PRINT : POKE 1659,1: REM 接通打印机
110  PRINT "----中文----"
120  PRINT "在中文状态下打印汉字"
130  PRINT
140  PRINT D$;"PR#1": REM  进入英文状态,
接通打印机
150  PRINT "---ENGLISH---"
160  PRINT "PRINT ENGLISH "
170  PRINT "ZHONG HUA XUE XI JI"
180  PRINT
190  PRINT D$;"PR#3"
200  PRINT : POKE 1659,5
210  PRINT "中华学习机"
220  POKE 1659,0
230  END
```

中华学习机在中文状态下打印速度较慢，若要提高打印速度，可以设法使机器在打印汉字时按中文状态工作，而打印其它字符时按英文状态工作.
为此，必须使程序按需要进行两种状态的转换

下面是一个实例.

```
----中文----
在中文状态下打印汉字

---ENGLISH---
PRINT ENGLISH
ZHONG HUA XUE XI JI

中华学习机
```

请观察打印输出的变化情况，体会打印中文汉字和英文字符时速度上的差异。

30句进入中文状态并接通打印机，将40—60句的中文信息打印出来，70句断开扫印机。90句又进入中文系统，100句接通打印机并按1号字型打印110—120句的中文汉字。140句进入英文状态，按打印英文的速度打印输出150—170句的字符。190句再次进入中文状态，按5号字型打印“中华学习机”五个汉字。220句是中文状态下断开打印机的命令。

2. 汉字打印控制

在中华学习机汉字系统中，汉字的打印驱动程序是针对MX-80Ⅲ型打印机设计的，它要求打印机的接口为Centronics接口，打字机接口必须定义为I/O槽口的1号槽。

进入汉字系统后，可以用下述四个软开关来控制和打印汉字。这些软开关 POKE 命令，既可以直接从键盘输入执行，也可以放在程序中使用。

(1) 设置打印方式

格式：POKE 1659，n

说明：n = 0为不打印，也表示脱离打印机。相当于西文状态的PR # 0功能。进入汉字系统后，打印方式自动置为0，n = 1—15为打印方式，中华学习机可以打印15种字型。

例如，　POKE 1659，1——打印小字

POKE 1659，2——横向小字打印

POKE 1659，3——横向大字打印

POKE 1659，5——纵向小字打印

POKE 1659， 6——纵向大字打印

（2）设置字间距

格式： POKF 1787，n

说明： n 为打印的 ASC II字符之间的字间距，即以英文字符为单位，英文字间隔为N个点，而一个汉字占用二个 ASC II字符的位置，所以中文字间隔为2N个点。n取值范围为0—255，字间距可在0—255个点之间。进入中文系统后，字间距自动置为1。

（3）设置行间距

格式：POKE 1915，n

说明：n 为打印行之间的点间距，n 取值在0—255之间。即定义行与行之间的距离为N个点，如果不加选择，进入系统后自动置为1。

（4）设置行允许字数

格式：POKE 2043，n

说明：n 为一行所允许打印的汉字个数。n 取值在0—255之间，由于一个汉字占用两个英文字符位置，所以每行允许打印的汉字数是N/2个。但实际上一行允许打印的汉字个数与所选择的打印方式，字型大小以及设定的字间距有关。

运行程序H44，可以进入汉字状态，并在打印机上用各种字体打印出中文字。读者可以输入并运行该程序，观察其执行结果。

应该特别指出的是，程序中软开关的设置状况发生转换时，应注意恢复，否则，打印的结果将发生错误，并发生打印机乱走纸和失控情况。

例如，220句用第6种字型，字与字之间间距30个点，打

印230句结果后，换第5种字型时，应将字间距恢复，即安排POKE1787，1再打印250句结果。

又如270句设置了行间距 POKE 1915，40后，在打印280句和290句结果后，转向另一种打印模式，应安排300句 POKE 1915，1，使其行间距恢复为正常模式。

其它情况，请体会程序中有关语句安排，并对照运行结果一一检验。

```
5 REM H44
10 REM 汉字打印控制
20 POKE 1659,5
30 PRINT "POKE1659,N:N--设置打印方式,N=0不打印,N=1--15为打印各种字型. 举例如下:"
40 PRINT
50 POKE 1659,1: REM 接通打印机,打印小字
60 PRINT "例1:N=1,各种字体中文打印一打印小字"
70 PRINT
80 POKE 1659,2: REM 打印横向小字
90 PRINT "例2:N=2,各种字体中文打印一横向小字"
100 PRINT
110 POKE 1659,3: REM  打印横向大字
120 PRINT "例3:N=3,各种字体中文打印一横向大字"
130 PRINT
140 POKE 1659,5: REM  打印纵向小字
150 PRINT "例4:N=5,各种字体中文打印一纵向小字"
160 PRINT
170 POKE 1659,6: REM  打印纵向大字
180 PRINT "例5:N=6,各种字体中文打印一纵向大字"
190 PRINT : POKE 1659,5
200 PRINT "POKE1787,N:N--设置字间距. N取0--255个点. 举例如下:"
210 PRINT
220 POKE 1659,6: POKE 1787,30
230 PRINT "各种字体中文打印"
240 PRINT : POKE 1659,5: POKE 1787,1
250 PRINT "POKE1915,N:N--设置行间距. N取0--255个点. 举例如下:"
```

```
260  PRINT
270  POKE 1659,6: POKE 1915,40
280  PRINT "各种字体中文打印"
290  PRINT "各种字体中文打印"
300  POKE 1915,1: PRINT : POKE 1659,5
310  PRINT "POKE2043,N:N--设置行允许字数. 取0--255个汉字. 举例如下:"
320  PRINT
330  POKE 1659,1: POKE 2043,48
340  PRINT "各种字体中文打印各种字体中文打印各种字体中文打印各种字体中文打印各种字体中文打印各种字体中文打印"
350  POKE 1659,0
360  END
```

POKE1659,N:N--设置打印方式,N=0不打印,N=1--15为打印各种字型. 举例如下:

例1:N=1,各种字体中文打印—打印小字

例2:N=2,各种字体中文打印—横向小字

例3:N=3,各种字体中文打印—横向大字

例4:N=5,各种字体中文打印—纵向小字

例5:N=6,各种字体中文打印—纵向大字

POKE1787,N:N--设置字间距. N取0--255个点. 举例如下:

各　　种　　字　　体　　中　　文　　打　　印

POKE1915,N:N--设置行间距. N取0--255个点. 举例如下:

各种字体中文打印

各种字体中文打印

POKE2043,N:N--设置行允许字数. 取0--255个汉字. 举例如下:

各种字体中文打印各种字体中文打印各种字体中文打印各种字体中文打印各种字体中文打印各种字体中文打印

3. 汉字图形混合打印

中华学习机可以利用自己的中文系统和高分辨率作图语句，方便地实现高分辨图形、汉字及字符的混合显示和打印。

由于中华学习机汉字系统是利用高分辨率图形页面2 来显示汉字和字符的，所以，只要在显示汉字的同时使HCOLOR 命令有效，便可在同一画面中混合输出汉字和图形。这实际上就是在高分辨图形中显示汉字和图形。

下面是一个简单的实例，在中文状态下，运行程序H45后，可以得到如图4.1的输出：

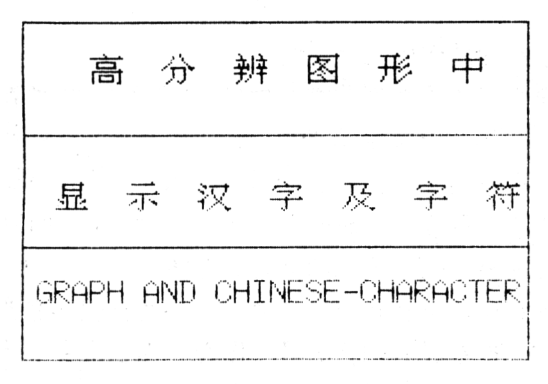


图 4.1

```
]LIST

2  REM H45
5  REM 汉字图形混合打印
10  HGR2
```

```
20 HOME : VTAB 2: HTAB 8: PRINT "高 分 辨
  图 形 中": VTAB 6: HTAB 6: PRINT "显 示
  汉 字 及 字 符"
25 VTAB 9: HTAB 5: PRINT "GRAPH AND CHINESE
-CHARACTER"
40 HCOLOR= 3: HPLOT 25,0 TO 250,0 TO 250,18
0 TO 25,180 TO 25,0: HPLOT 25,60 TO 249,60:
HPLOT 25,120 TO 249,120
50 K = PEEK ( - 16384): POKE ( - 16368),0:
IF K < 128 THEN 50
60 IF K = 208 THEN PR# 1: POKE 1913,2: PRI
NT CHR$ (17): END
70 PRINT CHR$ (18): PRINT CHR$ (12): END
```

程序编制和操作中几点说明：

（1）程序一开始必须设置一次HGR 2命令，以便在作图开始后使HCOLOR命令有效。

（2）本程序是在中文状态下运行，因此进入汉字显示方式后，由于显示区使用存储器高分辨第2页，故使用HGR 2命令不仅清屏幕，而且连同状态行也被清除。

（3）本程序不能使用HGR命令，因为它使屏幕变为混合显示方式，即在屏幕的上方显示图象，而在屏幕下方有4行字符的显示，但在这种方式下不能显示汉字。

（4）PRINT CHR$（12），可用在程序中清除屏幕；PRINT CHR$（18）的作用是显示或清除状态行的提示符。

（5）本程序停止按任意键，而打印则按P键。按P键后首先接通打印机（PR＃1），然后设置第二页打印方式单元

(POKE 1913，2)，最后打印(PRINT CHR$ (17))。

附：HGR2图形打印中POKE 1913，n含义：

POKE 1913，2：正常方式正常尺寸。

POKE 1913，66：放大尺寸图形。

POKE 1913，34：反相状态图形。

POKE 1913，98：反相放大图形。

地址1913打印方式单元的值一经定义后，除非用 POKE 语句重新设置，或重新开机，否则不受（ RESET ）等键的影响，始终保持已指定的打印方式。

4. 图形汉字硬拷贝

画好一幅图形，再加注一些中文说明，就能达到图文并茂的效果，从而使图象信息更为丰富，画面更加生动活泼。

程序H46，H47，H48是解决图形汉字硬拷贝的一些尝试，有的只能在屏幕上显示而不能全部打印；有的能显示也能打印但控制要有人的干预；有的比较好地解决了图形汉字的同时拷贝。

程序H46：

```
]LIST

5  REM H46
10 D$ =  CHR$ (4)
20  INPUT X
40  ON X GOSUB 100,900
50  GOTO 20
100  HGR2 : HCOLOR= 3
110  HPLOT 110,8 TO 110,50
```

```
120  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
130  HPLOT 80,100 TO 80,150
140  HPLOT 140,100 TO 140,150
230 X1$ = "【与 门 电 路 符 号】"
240  PRINT  TAB( 4);X1$
260  RETURN
900  END
```

运行程序H46后，在屏幕上显示高分辨第二页的图形和门电路四个汉字，见图4.2。

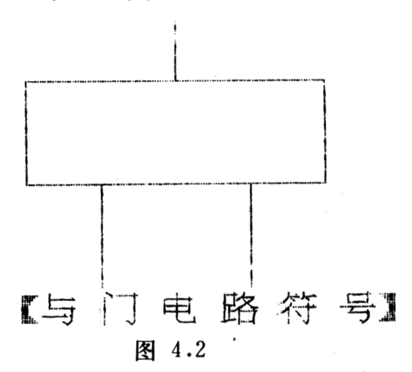


图 4.2

但是图 4.2 中图形和汉字并存的格式，却不容易拷贝下来。

在中文状态下接通打印机后，屏幕上先显示图形，然后只打印汉字而没有图形：

```
]RUN
?1
```

【与 门 电 路 符 号】

键入POKE1913，2和PRINT CHR$(17)后，图4.2中图形上半部打印不出来，见图4.3：

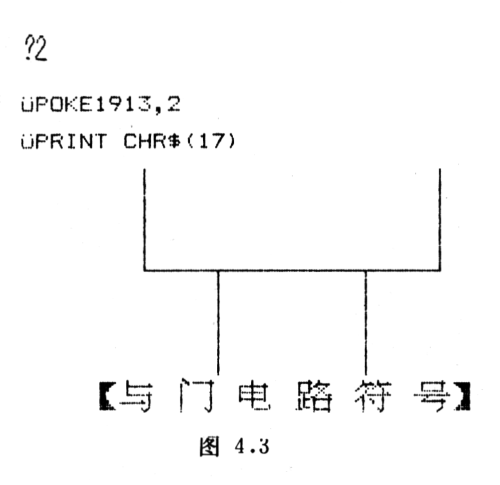


图 4.3

第一种情况，因没有设置高分辨图形打印方式，故打印输出只有与门电路四个汉字。第二种情况，虽设置了高分辨图形打印方式 POKE 1913,2 和打印图形指令 PRINT CHR$(17)，但图形的上半部没有打印出来。原因是在第一种情况执行后，没有安排 TEXT 命令，即没有及时从高分辨图形方式转换到满屏幕文本显示方式。

程序H47：

```
]LIST

5  REM H47
10 D$ =  CHR$ (4)
20  INPUT X
30  IF X = 3 THEN  END
40  ON X GOSUB 200,100,190
50  GOTO 20
100  HGR2 : HCOLOR= 3
110  HPLOT 110,8 TO 110,50
120  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
130  HPLOT 80,100 TO 80,150
140  HPLOT 140,100 TO 140,150
145  PRINT D$;"PR#1"
150  POKE 1913,2: PRINT  CHR$ (17)
155  PRINT D$"PR#3"
160  RETURN
190  END
200  PRINT
220  POKE 1659,5
230 X1$ = "【与 门 电 路 符 号】"
240  PRINT  TAB( 15);X1$
245  TEXT
250  POKE 1659,0
260  RETURN
```

程序H46的运行结果表明，在图形和中文信息均要打印的情况下，应该注意两种状态（中文状态和图形状态）的及时转换。

例如，打印中文信息后，安排TEXT语句，使其从图形状态（高分辨图形方式或低分辨图形方式）转换到满屏幕文本显示方式，而在打印图形后（这里指高分辨图形），用程序控制使其重新进入中文状态。即在执行PRINT CHR$(17)后，已经退出中文汉字系统状态，重新进入中文状态，必须用PR #3 命令，特别要注意的是，在该命令后一定要尾随一个PRINT语句。

程序H47就是这样安排的。为了说明原理，我们用一个总控来分别处理中文和图形两种状态的控制。

```
]RUN
?1
```

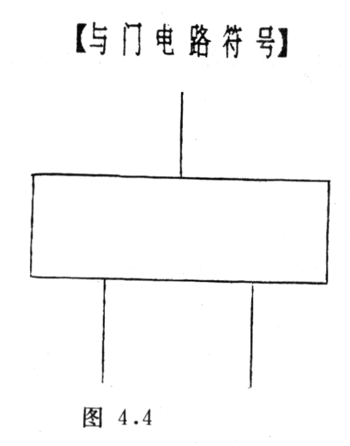
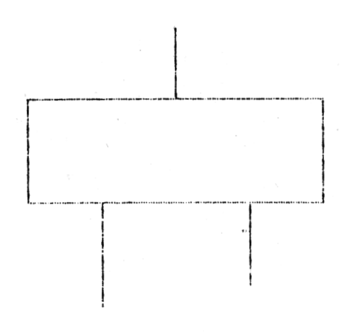


图 4.4

图4.4是先键入1，再键入 2 的打印结果。

图4.5是先键入2，再键入 1 的打印结果。

程序H47的优点是，键入1，或 2 的次序可以灵活掌握，既可以将中文加注在图形上方，也可以将中文加注在图形的



【与门电路符号】

图 4.5

下方，这是因为在程序安排了一种状态到另一种状态的及时切转。但缺点是一幅图形和文字的打印要分两步完成，并且要由人工操作进行。

为使操作简单，并不受人的干预，只要在程序H47的基础上，稍加改动部份语句和结构，即可自动完成图形汉字的硬拷贝工作。请看程序H48。

程序H48：

```
]LIST

5  REM H48
10 D$ =  CHR$ (4)
20  INPUT X
30  IF X = 2 THEN  END
40  ON X GOSUB 100
```

```
50  GOTO 20
100  HGR2 : HCOLOR= 3
110  HPLOT 110,8 TO 110,50
120  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
130  HPLOT 80,100 TO 80,150
140  HPLOT 140,100 TO 140,150
145  PRINT D$;"PR#1"
150  POKE 1913,2: PRINT  CHR$ (17)
155  PRINT D$"PR#3"
160  PRINT
220  POKE 1659,5
230 X1$ = "【与 门 电 路 符 号】"
240  PRINT  TAB( 15);X1$
245  TEXT
250  POKE 1659,0
260  RETURN
```

请注意，155句PRINT D$;“PR#3”的下续语句160句的PRINT是必不可少的，读者可以上机试验，删去160句后是什么情况，又是什么结果。

5．通用写字绘图程序

程序H49，既简单又通用，既可以写字又可以绘图。

```
]LIST
15 REM H49
18  PRINT
20  READ X: IF X < 0 THEN  PRINT : GOTO 20
```

```
30  IF X > 36 THEN 180
35  PRINT  TAB( X);" ";
40  READ Y
50  FOR I = X TO Y
60  PRINT "B";
70  NEXT I
80  GOTO 20
180  END
190  DATA 0,4,38
```

运行这个程序,可以写出由 5 个 B字符组成的“一”字。程序执行过程如下:

首先读一个数零，因不满足20句和30句的条件而转向35句，在X＝0的位置上打印一个空格，实际上是控制打印针头处在起始位置。执行40句读一个数4，经过50—70句循环，连续打印 5 个“B”字符（因为X＝0，Y＝4），循环结束经80句再转向20句,又读一个值38,满足30句条件，转向 180 句然后结束。

如果在 DATA语句中安排小于零的数，如－1，则满足20句条件打印一个空格，从而起到控制打印针头换行的作用。

因此，我们事先写好字或画好图（最好用方格坐标纸），把数据放在DATA语句中，则可以利用上述程序，自动完成写字或画图的工作。

下面是一个实例，运行程序H50可以画出一个熊猫图案来。

```
]RUN

                      BBB
          BBBB BBBBB BBBBB
         BBBBBBB   BBBBBBB
         BBBBBB      BBBBB
        BBBBBBB       BBB
       BBBB            BB
      BBBB             BB
     BBBB        BBBB    BBB
    BBBB       BBBBB BBB BB
    BBBB       BBBBB BBB BB
    BBBB      BBBBBB BBBB BB
   BBBB       BBBBB   BBB BB
   BBBB      BBBBB       BB
  BBBBBB      BBB        BB
  BBBBBBB                BB
  BBBBBBBBB        BBB BBB
  BBBBBBBBBBBB       BBBBBBB
 BBBBBBBBBBBBBB     BBBBBBBBB
 BBBBBBBBBBBBBBBBBBBBBBBBBBBB
 BBBBBBBBBBBBBBBBBBBBBBBBBBBB
 BBBBBBBBBBBBBBBBBBBBBBBBBBBB
 BBBBBBBBBBBBBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBBBBBBBBBBBBBB
BBBBB  BBBBBBBB BBBBB BBB BBBB
BBBB   BBBBB  BBBBB  BB   BBBB
BBBB        BBBBBB  BBBB  BBBB
BBBBBB      BBBB   BBBBB  BBBB
BBBBBBBBB BBBBBB  BBBBBBBB BBBB
BBBBBBBBBBBBBBBB  BBBBBBBB BBBB
BBBBBBBBBBBBBBBB  BBBBBBBB BBBB
BBBBBBBBBBBBBBBB  BBBBBBB  BBBB
BBBBBBBBBBBBBBBB   BBBBBB  BBBB
BBBBBBBBBBBBBBBB   BBBBBB  BBBB
BBBBBBBBBBBBBBBBB   BBBB  BBBB
BBBBBBBBBBBBBBBBBB  BBB  BBBBB
 BBBBBBBBBBBBBBBBBB BBB BBBBB
    BBBBBBBBBBBBBBBB BBB BBBBB
    BBBBBBBBB   BBBBBBBBBBB
                 BBBBBBBBB
                  BBBBBBB
                    BBB
```

```
]LIST

5  REM H50
10  REM 熊猫程序
18  PRINT
20  READ X: IF X < 0 THEN  PRINT : GOTO 20
```

```
30  IF X > 36 THEN 150
35  PRINT  TAB( X);" ";
40  READ Y
50  FOR I = X TO Y
60  PRINT "B";
70  NEXT I
80  GOTO 20
150  RESTORE
180  END
200  DATA 23,25,-1,11,14,16,20,22,26,-1,10,1
6,20,26,-1,10,15,22,26,-1,9,15,23,25,-1
210  DATA 8,11,24,25,-1,7,10,24,25,-1,6,9,17
,20,25,27,-1,5,8,16,20,22,24,26,27,-1
220  DATA 5,8,16,20,22,24,26,27,-1,5,8,15,20
,22,25,26,27,-1,4,7,15,19,23,25,26,27,-1
230  DATA 4,7,14,18,25,26,-1,3,8,15,17,25,26
,-1,3,9,25,26,-1,3,11,20,22,24,26,-1
240  DATA 3,14,22,28,-1,2,15,21,29,-1,2,29,-
1,2,29,-1,2,29,-1,2,28,-1
250  DATA 1,27,-1,1,28,-1,1,5,8,15,16,20,22,
24,26,29,-1,1,4,8,12,15,19,22,23,26,29,-1
260  DATA 1,4,13,18,21,24,27,30,-1,1,6,13,16
,20,24,27,30,-1,1,9,11,16,19,26,28,31,-1
270  DATA 1,16,19,26,28,31,-1,1,16,19,26,28,
31,-1,1,16,19,25,28,31,-1,1,16,20,25,28,31,-
1,1,16,20,25,28,31,-1
280  DATA 1,17,21,24,27,30,-1,1,18,21,23,26,
30,-1,2,19,21,23,24,28,-1
290  DATA 5,20,21,23,24,28,-1,5,13,17,27,-1,
18,26,-1,19,25,-1, 21,23,-1,999
```

程序H51也可以写字或者画图，其设计思想和程序H50十分相似，M＝N＝0是起换行作用，M＝N＝36是终止程序的标志。运行程序H51后可以打印出＋、－、×、÷、＝五个数学符号。

程序中M的取数是程序中要求打印图形的起始位置，N的取数则是程序中要求打印符号(例如＊)的个数。一排排打印，直到M、N的值是36，则说明图形已经全部打完，当M、N的值均为“0”时，程序安排机器回车，表示一行打印完毕，另从下行开始打印，反复循环。

```
]LIST

5  REM H51
10  READ M,N
20  IF N = 36 THEN 100
30  IF M = 0 THEN 80
40  FOR I = M TO M + N - 1
50  PRINT  TAB( I);"*";
60  NEXT I
70  GOTO 10
80  PRINT
90  GOTO 10
100  END
110  DATA 5,1,17,1,21,1,26,1,0,0
120  DATA 5,1,18,1,20,1,31,5,0,0
130  DATA 3,5,10,5,19,1,24,5,0,0
140  DATA 5,1,18,1,20,1,31,5,0,0
150  DATA 5,1,17,1,21,1,26,1,0,0
160  DATA 36,36
```

]RUN



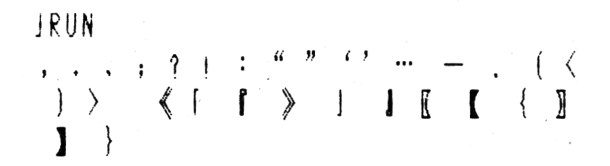
6. 打印特殊字符库

中华学习机在进入汉字系统后，按下“F2”键时，屏幕左下方提示行将显示“拼音:”两字,表示进入拼音输入方式。

键入－(减号)、＝(等号)、＼(斜线)可以选择标点符号、算术运算符号和制表线符号等。调用这些符号给程序编制、报表输出带来许多方便，并使报表形式更加多样、美观。

这些特殊符号的输入方法如下：

按 －(减号) 可输入的符号有：



按 ＝(等号) 可输入的符号有：

￥ % + − × × ÷ ± = ≈ > < > < / (
[ )] ∑ ~ ① ② ③ ④ ⑤ ⑥ ⑦
⑧ ⑨ ⑩

按＼（斜线）可输入的符号有：

─ │ ┌ ┐ └ ┘ ├ ┤ ┬ ┴ ┼

汉字系统中所提供的特殊符号，远远不止以上71种，事实上还有好几百个符号，例如，下面一些特殊符号：

ö ò û ù ÿ Ö Ü ¢ £ ¥
¿ ⌐ ¬ ½ ¼ ¡ « » ▒ ▓
╝ ╜ ╛ ┐ └ ┴ ┬ ├ ─ ┼
╨ ╤ ╥ ╙ ╘ ╒ ╓ ╫ ╪ ┘
Σ σ µ τ Φ Θ Ω δ ∞ ø

₧ ƒ á í ó ú ñ Ñ ª º
░ │ ┤ ╡ ╢ ╖ ╕ ╣ ║ ╗
╞ ╟ ╚ ╔ ╩ ╦ ╠ ═ ╬ ╧
┌ █ ▄ ▌ ▐ ▀ α β γ π
∈ ∩ ≡ ±

☺ ☻ ♥ ♦ ♣ ♠ • ◘ ○
¶ § ▬ ↨ ↑ ↓ → ← ∟ ↔
. ∀ % $ # .. ! ▲ ▼
‼ ↕ ► ◄ ☼ ♫ ♪ ♂ ♀ ◙

ㄫ ² ▮ ˋ ˙ ˙ ˉ ˇ

” ( ) ＜ ＞ 《 》 「 」 『

∨ ∑ ∏ ∪ ∩ ∈ ∷ √ ⊥ ∥

≠ ≮ ≯ ≤ ≥ ∞ ∵ ∴ ♂ ♀

№ ☆ ★ ○

≥ ≤ ⌠ ⌡ ÷ ≈ ° · ‐ ⌐

¨ 〃 々 — ～ ‖ … ‘ ’ “

』 〔 〕 【 】 ± × ÷ ∶ ∧

∠ ⌒ ⊙ ∫ ∮ ≡ ≌ ≈ ∽ ∝

° ′ ″ ℃ ＄ ¤ ￠ ￡ ‰ §

● ◎ ◇ ◆ □ ■ △ ▲ ※ →

11. 12. 13. 14. 15. 16. 17. 18. 19. 20.

⑾ ⑿ ⒀ ⒁ ⒂ ⒃ ⒄ ⒅ ⒆ ⒇

㈠ ㈡ ㈢ ㈣ ㈤ ㈥ ㈦ ㈧

← ↑ ↓ 〓

1. 2. 3. 4. 5. 6. 7. 8. 9. 10.

⑴ ⑵ ⑶ ⑷ ⑸ ⑹ ⑺ ⑻ ⑼ ⑽

① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩

㈨ ㈩

这些特殊字符存在什么地方，又怎样调出来呢？这和汉字按区位方式存贮在字库中十分类似。

例如，国标中规定16区01位置上是汉字的“啊”字，在汉字系统状态下，按“F3”键后，屏幕左下方提示行将出现“区位:”两字，表示进入区位码输入方式。此时键入1601后，“啊”字就输入到计算机中，显示在屏幕上，即：

区 位 码		汉 字
16	01	啊
区号	位号	

同样，汉字“啊”国际码为1601，它在中华学习机内的表示形式为：7F 2E 1D。这就是说每个汉字均以等长的3字码保存在主机内存中，其格式为：7F＋区码＋位码。

其中7F是定值，2E是区码，1D是位码。2E是学习机内码的16进制表示，对应的学习机内码10进制表示则为46 ($2\times16+14\times16^{\circ}=46$)，相应的区位码的10进表示则为16；1D是学习机内码的16进制表示，对应的学习机内码10进制表示则为29 ($1\times16+13\times16^{\circ}=29$)，相应的区位码的10进制表示则为01；而7F也是16进制表示，化成10进制数应为127 ($7\times16+15\times16^{\circ}=127$)。

因此，汉字“啊”，也可以用以下形式找出来，则从键盘输入：PRINT CHR$(127)+CHR$(46)+CHR$(29)，屏幕上立即显示“啊”。

CHR$(算术表达式）是一个函数，它的作用是得到一个与算术表达式值相对应的ASC II字符。例如，PRINT CHR$(65)，打印出“A”字符来。

汉字“啊”字，在BASIC程序中也可以用字符串函数表示，如 A$ = CHR$(127)+ CHR$(46)+ CHR$

(29)，则PRINT A$也会显示出汉字“啊”字。

有了上述基本知识，我们可以用一个程序将学习机内所有的特殊字符全部打印出来。

关键是必须知道,这些特殊字符是存贮在那一个地址,亦即它们的区号在什么地方。经对中华学习机汉字库与ASC II字串的分析，了解到它们存贮在122,123,124,125,126 这五个区号中，而各自的位号为 1 到94。

打印特殊字符库的程序见H52：

```
]LIST

1  REM H52
2  REM  打印特殊字符库
5  POKE 2043,50
7  POKE 1787,4
8  POKE 1915,4
9  POKE 1659,1
10  FOR I = 76 TO 80
12  PRINT  TAB( 10);
15  PRINT "CHR$(127)+CHR$(";46 + I;")";"+CHR
$(J)";
18  PRINT : PRINT
20  FOR J = 0 TO 96
28 X = 7 * (K + 1) - A
30  PRINT  TAB( X);29 + J;"-";
40  PRINT  CHR$ (127) +  CHR$ (46 + I) +  CH
R$ (29 + J);
50 K = K + 1
60  IF X > = 34 THEN  PRINT :K = 0:X = X -
34
```

```
80  NEXT J
90  FOR S = 1 TO 25: PRINT : NEXT S
95 K = 0:A = 0
100  NEXT I
105  POKE 1659,0
110  END
```

程序H52的运行结果：

```
    CHR$(127)+CHR$(122)+CHR$(J)

29-     30-     31-     32-     33-
34-     35-     36-     37-     38-
39-     40-     41-     42-     43-
44-     45-     46-     47-     48-
49-     50-     51-     52-     53-
54-     55-     56-     57-     58-
59-     60-     61-     62-     63-
64-     65-     66-     67-     68-
69-     70-     71-     72-     73- ☻
74- ●   75- ♥   76- ♦   77- ♣   78- ♠
79- •   80- ◘   81- ○   82- █   83- ♂
84- ♀   85- ♪   86- ♫   87- ☼   88- ►
89- ◄   90- ↕   91- ‼   92- ¶   93- §
94- ▬   95- ↨   96- ↑   97- ↓   98- →
99- ←   100- ∟  101- ↔  102- ▲  103- ▼
104-    105- !  106- "  107- #  108- $
109- %  110- &  111- '  112- (  113- )
114- *  115- +  116- ,  117- -  118- .
119- /  120- 0  121- 1  122- 2  123- 3
124- 4  125- 5
```

CHR$(127)+CHR$(123)+CHR$(J)

29-6 30-7 31-8 32-9 33-:
34- 35-; 36-< 37-= 38->
39-? 40-@ 41-A 42-B 43-C
44- 45-D 46-E 47-F 48-G
49-H 50-I 51-J 52-K 53-L
54-M 55-N 56-O 57-P 58-
59-Q 60-R 61-S 62-T 63-U
64-V 65-W 66-X 67-Y 68-Z
69-[70-\ 71-] 72-^ 73-_
74-` 75-a 76-b 77-c 78-d
79-e 80-f 81-g 82-h 83-i
84-j 85-k 86-l 87-m 88-n
89-o 90-p 91-q 92-r 93-s
94-t 95-u 96-v 97-w 98-x
99-y 100-z 101-{ 102-¦ 103-}
104-~ 105-⌂ 106-Ç 107-ü 108-é
109-â 110-ä 111-à 112-å 113-ç
114-ê 115-ë 116-è 117-ï 118-î
119-ì 120-Ä 121-Å 122-É 123-æ
124-Æ 125-ô

CHR$(127)+CHR$(124)+CHR$(J)

29-ö 30-ò 31-û 32-ù 33-ÿ
34- 35-Ö 36-Ü 37-¢ 38-£
39-¥ 40-₧ 41-ƒ 42-á 43-í
44- 45-ó 46-ú 47-ñ 48-Ñ
49-ª 50-º 51-¿ 52-⌐ 53-¬
54-¼ 55-½ 56-¡ 57-« 58-
59-» 60-▒ 61-█ 62-█ 63-│
64-┤ 65-╡ 66-╢ 67-╖ 68-╕
69-╣ 70-║ 71-╗ 72-╝ 73-╛
74-┘ 75-┐ 76-└ 77-┴ 78-┬
79-├ 80-─ 81-┼ 82-╞ 83-╟

84 - ┗	85 - ┏	86 - ┻	87 - ┳	88 - ┣
89 - ━	90 - ╋	91 - ┴	92 - ┻	93 - ┳
94 - ┳	95 - ┖	96 - └	97 - ┌	98 - ┏
99 - ╂	100 - ┼	101 - ┘	102 - ┌	103 - █
104 - ■	105 - ▌	106 - ▌	107 - ▀	108 - ∝
109 - β	110 - γ	111 - π	112 - Σ	113 - σ
114 - μ	115 - τ	116 - Φ	117 - θ	118 - Ω
119 - δ	120 - ∞	121 - ∅	122 - ∈	123 - ∩
124 - ≡	125 - ±			

CHR$(127)+CHR$(125)+CHR$(J)

29 - ≥	30 - ≤	31 - ┌	32 - ┘	33 - ÷
34 -	35 - ≈	36 - •	37 - ·	38 - ·
39 - ɾ	40 - ⁿ	41 - ▪	42 - ▮	43 -
44 -	45 -	46 - 、	47 - .	48 - ·
49 - ¯	50 - ˇ	51 - ¨	52 - 〃	53 - 々
54 - —	55 - ～	56 - ‖	57 - …	58 -
59 - ‘	60 - ’	61 - “	62 - ”	63 - (
64 -)	65 - 〈	66 - 〉	67 - 《	68 - 》
69 - 「	70 - 」	71 - 『	72 - 』	73 - 〖
74 - 〗	75 - 【	76 - 】	77 - ±	78 - ×
79 - ÷	80 - :	81 - ∧	82 - ∨	83 - Σ
84 - Π	85 - ∪	86 - ∩	87 - ∈	88 - ∷
89 - √	90 - ⊥	91 - ∥	92 - ∠	93 - ⌒
94 - ⊙	95 - ∫	96 - ∮	97 - ≡	98 - ≌
99 - ≈	100 - ∽	101 - ∝	102 - ≠	103 - ≮
104 - ≯	105 - ⩽	106 - ⩾	107 - ∞	108 - ∵
109 - ∴	110 - ♂	111 - ♀	112 - °	113 - ′
114 - ″	115 - ℃	116 - $	117 - ¤	118 - ¢
119 - £	120 - ‰	121 - §	122 - №	123 - ☆
124 - ★	125 - ○			

```
CHR$(127)+CHR$(126)+CHR$(J)

29-●     30-○     31-◇     32-◆     33-□
34-      35-■     36-△     37-▲     38-※
39-→     40-←     41-↑     42-↓     43-═
44-      45-      46-      47-      48-
49-      50-      51-      52-      53-
54-      55-      56-      57-      58-
59-      60-      61-      62-1.    63-2.
64-3.    65-4.    66-5.    67-6.    68-7.
69-8.    70-9.    71-10.   72-11.   73-12.
74-13.   75-14.   76-15.   77-16.   78-17.
79-18.   80-19.   81-20.   82-(1)   83-(2)
84-(3)   85-(4)   86-(5)   87-(6)   88-(7)
89-(8)   90-(9)   91-(10)  92-(11)  93-(12)
94-(13)  95-(14)  96-(15)  97-(16)  98-(17)
99-(18)  100-(19) 101-(20) 102-①   103-②
104-③   105-④   106-⑤   107-⑥   108-⑦
109-⑧   110-⑨   111-⑩   112-     113-
114-㈠   115-㈡   116-㈢   117-㈣   118-㈤
119-㈥   120-㈦   121-㈧   122-㈨   123-㈩
124-     125-
```

关于“打印特殊字符库”程序及运行结果的几点说明：

① I 循环变量从76到80，这样46＋I 分别为122到 126，相 应于 91 区到95 区。 J 循环变量从 0 到96，这样29＋ J 对应于 0 —94个位号。

② 5 个POKE 指令，分别为设置行最大允许字符个数，字间距，行间距，接通打印机按 6 号字型打印以及关闭打印机。

③ 15句为打印一个表头，其形式为：

CHR $(127)＋CHR $(46＋I)＋CHR $(J)

如 I =76时，则上式变为：

CHR$(127)+CHR$(122)+CHR$(J)

30句是打印位号和一个“-”号，所以运行结果中每一个特殊符号前的数值(29-125)是位号。

40句打印的是相应区号和位号的特殊字符。若在位号后面没有特殊符号打印出来，则说明系统内部在该区位号没有存贮特殊字符。

④ TAB (X)是打印定位，开始X = 7 * (K+1) - A，因K和A开始均未赋值，X=7，即最初从第7号位置开始打印结果。利用K=K+1作计数题，以后各项空7个标准位置打印。

考虑到汉字状态下，系统以17个汉字（即34个字符）为模，所以在X>=34后应减去34，否则打印位置混乱。

在打印全部位号的特殊符号后，到换另一个开始时，将指针恢复即K=0，A=0。

⑤ 90句安排了一个空循环，控制打印纸空25行，这是为打印好一个区的所有特殊字符后，安排的跳页处理，这样另一个区的94个特殊字符将打印在另一页新纸上，使输出结果比较美观，便于资料保存。

⑥ 这些特殊字符的调用方法：只要查运行结果中区位号。例如要调用Σ符号，它的区号是124，位号为112，故可以用以下方法处理：

```
10. A$ =CHR$(127)+CHR$(124)
        +CHR$(112)
```

这样就将Σ符号赋给字符串变量A$中，而打印输出时用：

```
20  PRINT   A$
```

```
]POKE2043,24
]LIST
1  REM  区位字符查找
4  POKE 2043,22
6  POKE 1787,1
8  POKE 1915,3
10  REM 中华学习机 CEC-I
25  PRINT : PRINT
30  REM "输入区号:1-9"
40  FOR A = 1 TO 9
45  PRINT : PRINT
50  PRINT  TAB( 20);"第 ";A;" 区"
55  PRINT
60 J = 0
70 F = A: GOSUB 500
80 Q = FF
90  FOR B = 1 TO 94
100 F = B: GOSUB 500
110 W = FF
120 A$ =  CHR$ (127) +  CHR$ (Q) +  CHR$ (W)
130  IF B < 10 THEN B$ = "0" +  STR$ (B): GOTO 1
50
140 B$ =  STR$ (B)
150  IF A < 10 THEN AA$ = "0" +  STR$ (A): GOTO
170
160 AA$ =  STR$ (A)
170 X = 7 * (K + 1)
174  PRINT  TAB( X);
```

```
176  PRINT AA$;B$;A$;
180 J = J + 1: IF J = JJ THEN  PRINT :J = 0
185 K = K + 1
187  IF X > 34 THEN  PRINT :K = 0:X = X - 34
190  NEXT B
192  NEXT A
195  PRINT : PRINT : PRINT
200  END
500  IF F > 0 AND F < 6 THEN FF = 28 + F
510  IF F > 5 AND F < 15 THEN FF = 29 + F
520  IF F > 14 AND F < 28 THEN FF = 30 + F
530  IF F > 27 AND F < 95 THEN FF = 31 + F
540  RETURN
```

为便于查找全部中华学习机字符区位码，请用区位字符查找程序，它的运行结果请参见附录 1 中华学习机字符区位码。

7. 特殊符号应用

中华学习机汉字系统中，除了存贮国标一、二级汉字外，还存贮了好几百个特殊符号，如各种标点符号，算术运算符号，制表线符号以及一些特殊造型符号。调用这些符号，会给程序编制，报表输出带来不少方便，并使画面、输出形式更加丰富多采。

现举三个应用实例。

例 1，设计一个程序，要求打印以下数学表达式，并得出计算结果：

```
]RUN
       6      ½     ¼    ²    ½     ¼
   S = Σ 1 +4  -16  +32  -64  +128
      j =1
   S = 1020.36359
```

对于以上计算结果，任何一种型号的微机都能完成，但对上述数学表达式的打印输出，却不是每一种计算机都能完成的。

为了解决上述数学表达式的打印，必须调用中华学习机内的特殊字符。如Σ，小写的1/2，1/4，2等。

方法是从特殊字符库中，查找有关符号的区位码值，并赋给字符串变量，然后在适当的地方打印，只要注意定位即可。

如，A$(1)=CHR$(127)+CHR$(124)+CHR$(112)，就是将Σ存在A$(1)中，本程序中其它几个特殊字符分别存在下列单元中：

A$(2)：j

A$(3)：1/2

A$(4)：1/4

A$(5)：2

以上字符均为小写，它们可以在"打印特殊字符库"一文中找到。

下面是用POKE1659,1 指令打印出来的1号字型的结果：

```
]RUN
        6     ½    ¼    *    ½    ¼
   S = Σ 1 +4  -16  +32  -64  +128
     j =1

   S = 1020.36359
```

程序见特殊符号应用一：

```
]LIST

5  REM 特殊符号应用一
8  DIM A$(10)
10 A$(1) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (112)
12 A$(2) =  CHR$ (127) +  CHR$ (123) +  CHR$
 (84)
14 A$(3) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (54)
16 A$(4) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (55)
18 A$(5) =  CHR$ (127) +  CHR$ (125) +  CHR$
 (41)
40  PRINT  TAB( 8);"6"; TAB( 14);A$(3); TAB(
 18);A$(4); TAB( 22);A$(5); TAB( 26);A$(3);
TAB( 31);A$(4)
50  PRINT  TAB( 4);"S = ";A$(1);"1 +4 -16 +3
2 -64 +128 "
60  PRINT  TAB( 6);A$(2);"=1"
70 S = 1 +  SQR (4) -  SQR ( SQR (16)) + 32
* 32 -  SQR (64) +  SQR ( SQR (128))
75  PRINT
80  PRINT  TAB( 4);"S = ";S
90  END
```

请注意本程序中特殊字符，一定要放在字符串变量中。它不能放在REM，PRINT"……"，INPUT "……"，DATA等语句中。而所有汉字则可以放在以上各种形式的语句中。

程序中为使打印格式符合题意要求，一定要细心调整TAB(X)中的X值。

整个一个数学表达式是分三行处理的，它们分别由40句、50句、60句完成。

例2，设计一个程序，打印如下表头：

```
]RUN

┏ - - - - - - - - - - - - - - - - - ┓
┃  中 华 学 习 机 编 程 技 巧        ┃
┃          朱 国 江                 ┃
┃        气 象 出 版 社             ┃
┗ - - - - - - - - - - - - - - - - - ┛
```

这是一个普通的表头，打印出来不会有太多的困难，主要是调用中华学习机内的特殊字符，正如前面指出的，表头中的一些制表线符在一些微机中是没有的。

存贮单元设置：

A$(0)：‖

A$(1)：【本程序中未用

A$(2)：】本程序中未用

A$(3)：中华学习机编程技巧

A$(4)：朱国江

A$(5)：气象出版社

A $(6)：╔

A $(7)：═

A $(8)：╗

A $(9)：╚

A $(10)：╝

程序简要说明：

制表符"═"，用的较多，本程序不采用字符串原样打印格式，而采用循环方法处理，即：

FOR I =1 TO 14:PRINT A $(7);:NEXT I

这样可以连续打印14个"═"符号。

制表符"‖"，采用两式形式处理，没有汉字的行，采用调子程序的方法打印，即：

```
50   GOSUB  200
200  PRINT  A $(0);"␣␣␣␣ ......";A $(0)
210  RETURN
```

而在有汉字的行，采用TAB(X)函数处理。

其余制表符采用TAB(X)形式打印。

程序见特殊符号应用二：

```
]LIST

3  PRINT : PRINT
5  REM 特殊符号应用二
8  DIM A$(10)
10 A$(1) =  CHR$ (127) +  CHR$ (125) +  CHR$
 (75)
12 A$(2) =  CHR$ (127) +  CHR$ (125) +  CHR$
 (76)
14 A$(3) = "中 华 学 习 机 编 程 技 巧"
```

```
16 A$(4) = "朱 国 江"
18 A$(5) = "气 象 出 版 社"
20 A$(6) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (85)
22 A$(7) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (89)
24 A$(8) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (71)
26 A$(9) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (84)
28 A$(10) =  CHR$ (127) +  CHR$ (124) +  CHR
$ (72)
30 A$(0) =  CHR$ (127) +  CHR$ (124) +  CHR$
 (70)
44  PRINT A$(6);
46  FOR I = 1 TO 14: PRINT A$(7);: NEXT I
48  PRINT A$(8);: PRINT
49  PRINT A$(0); TAB( 4);A$(3);" ";A$(0)
50  GOSUB 200
52  PRINT A$(0); TAB( 13);A$(4);"            "
;A$(0)
55  GOSUB 200
60  PRINT A$(0); TAB( 10);A$(5);"        ";A$
(0)
70  PRINT A$(9);
80  FOR I = 1 TO 14: PRINT A$(7);: NEXT I
90  PRINT A$(10)
100  END
200  PRINT A$(0);"
     "; A$ (0 )
210  RETURN
```

例 3，编制一个程序，要求用特殊符号★作为表头的边框，格式如下：

]RUN



编制方法同例 2，利用输出空格函数(SPC)也很自由灵便，程序见特殊符号应用三：

```
]LIST

3  PRINT : PRINT
5  REM  特殊符号应用三
8  DIM A$(5)
10 A$(1) =  CHR$ (127) +  CHR$ (125) +  CHR$
 (124)
14 A$(3) = "中 华 学 习 机 汉 字 软 件"
16 A$(4) = "朱 国 江"
18 A$(5) = "气 象 出 版 社"
46  GOSUB 200
48  PRINT
49  GOSUB 150
50  PRINT A$(1); TAB( 4);A$(3);" ";A$(1)
60  GOSUB 150
```

```
70  PRINT  SPC( 0);A$(1); SPC( 10);A$(4); SP
C( 10);A$(1)
80  PRINT A$(1); TAB( 10);A$(5);"       ";A$
(1)
100  GOSUB 200
105  PRINT
110  END
150  PRINT  SPC( 0);A$(1); SPC( 28);A$(1)
160  RETURN
200  FOR I = 1 TO 16: PRINT A$(1);: NEXT I
210  RETURN
```

五、实 用 软 件

1. 销售情况统计

已知某电脑公司 1 至 4 月份销售计算机情况如表5.1。

表 5.1

型号 / 单价 / 月份	IBM/PC	长城0520	中华机
单价	31000	28600	1000
一	110	120	130
二	140	150	160
三	200	210	220
四	230	240	250

试编制一个销售统计程序。

程序可以分四段设计:

(1) 销售计算机情况

用二维数组比较方便,见程序中的15—80句。对于 I 个月份, J 种机型的销售台数15—80 句是一个通用模式。

(2) 计算机价格

因为只有三种价格, 安排一重循环即可, J 为循环变量, 价格放在 S(J)中, J =1,2,3。

(3) 营业额计算

用二重循环处理, 营业额放在T数组的T(1),T(2) 和 T(3),T(4)中, 分别代表每月销售额, 而全公司四个月份

的总营业额放在S中。

(4) 查找营业额

用键盘输入月份，直接打印该月份对应的T（I)值即为所求，I＝1,2,3,4。I ＝ 0时用条件语句判断打印S值。

现在来写程序：

首先把计算机出售的数量放在200—210句的 DATA 语句中。而把计算机的单价放在220句中。

用20—120句给数组A和数组S赋值,并打印出这两个数组的值。

130—190句计算营业额，并打印出结果。

250—260句为查找营业额，并打印结果。

完整的程序（见H 53）及运行结果：

```
]LIST

1  REM H53
2  REM   销 售 统 计
5  REM A(4,3)放销售量,S(3)放价格,T(4)中放营
业额
10  DIM A(4,3),S(3),T(4)
15  PRINT "销售计算机情况"
17  PRINT
20  FOR I = 1 TO 4
30  FOR J = 1 TO 3
40  READ A(I,J)
50  PRINT A(I,J);"  ";
60  NEXT J
70  PRINT
80  NEXT I
```

```
81  PRINT
82  PRINT "计算机价格"
85  PRINT
90  FOR J = 1 TO 3
100  READ S(J)
110  PRINT S(J);"  "
120  NEXT J
125  PRINT
127  PRINT "销售总营业额"
128 S = 0
129  PRINT
130  FOR I = 1 TO 4
140 T(I) = 0
150  FOR J = 1 TO 3
160 T(I) = A(I,J) * S(J) + T(I)
170  NEXT J
180  PRINT "T(";I;")=";T(I)
185 S = S + T(I)
190  NEXT I
192  PRINT
195  PRINT "公 司 总 额";" ";"S=";S
200  DATA 110,120,130,140,150,160
210  DATA 200,210,220,230,240,250
220  DATA   31000,28600,1000
230  REM  查找各月营业总额
235  PRINT
250  INPUT I
255  IF I = 0 THEN  PRINT "S=";S: GOTO 280
260  PRINT I;" 月 份 销 售 总 额";"  ";T(I):
 GOTO 250
280  END
```

```
]RUN
销售计算机情况

110  120  130
140  150  160
200  210  220
230  240  250

计算机价格

31000
28600
1000

销售总营业额

T(1)=6972000
T(2)=8790000
T(3)=12426000
T(4)=14244000

公 司 总 额 S=42432000

?1
1 月 份 销 售 总 额  6972000
?2
2 月 份 销 售 总 额  8790000
?3
3 月 份 销 售 总 额  12426000
?4
4 月 份 销 售 总 额  14244000
?0
S=42432000
```

几点说明：

（1）请注意20—80句的格式安排，50句最后面的分号“；”不能少，70句的PRINT不能省，否则打印出的结果不是一列，就是一行。

（2）140句T（I）=0，和128句S=0不能省，否则再次运行本程序会发生计算错误。

（3）本程序具有通用性，既可以统计营业额，也可以统计产量或者产值；既可以统计一个单位不同月份的几种产品，也可以统计几个单位的若干种产品。而只要改动I，J值并赋给变量A（I，J）以新的内容。

（4）为使本程序对增加数据有更多的灵活性，可以将I，J的终值加大，例如I=12，J=7，则表示可以处理12个月份，7种机型的销售情况，当然，DATA语句的信息也应相应增加。

（5）本程序还可以进一步完善，例如月份注上汉字，机型注上英文或中文名称；再加上表格，会使输出格式更为美观。这项工作留给有兴趣的读者完成。

下面提供一个参考程序，只要将原程序280句的END删去，改成280PRINT，然后加300—410句的程序段，即可打印一个计算机销售情况的表格。若需要打印表格输入I=0即可。完整的程序清单（见H53-1）及运行结果。

```
2  REM H53-1
5  REM A(4,3)放销售量,S(3)放价格,T(4)中放营
业额
10  DIM A(4,3),S(3),T(4)
```

```
15  PRINT "销售计算机情况"
17  PRINT
20  FOR I = 1 TO 4
30  FOR J = 1 TO 3
40  READ A(I,J)
50  PRINT A(I,J);"  ";
60  NEXT J
70  PRINT
80  NEXT I
81  PRINT
82  PRINT "计算机价格"
85  PRINT
90  FOR J = 1 TO 3
100  READ S(J)
110  PRINT S(J);"  "
120  NEXT J
125  PRINT
127  PRINT "销售总营业额"
128 S = 0
```

```
129  PRINT
130  FOR I = 1 TO 4
140 T(I) = 0
150  FOR J = 1 TO 3
160 T(I) = A(I,J) * S(J) + T(I)
170  NEXT J
180  PRINT "T(";I;")=";T(I)
185 S = S + T(I)
190  NEXT I
```

```
192  PRINT
195  PRINT "公 司 总 额";" ";"S=";S
200  DATA 110,120,130,140,150,160
210  DATA 200,210,220,230,240,250
220  DATA   31000,28600,1000
230  REM  查找各月营业总额
235  PRINT
250  INPUT I
255  IF I = 0 THEN  PRINT "S=";S: GOTO 280
260  PRINT I;" 月 份 销 售 总 额";"  ";T(I):
 GOTO 250
280  PRINT
300  PRINT "┌───┬───┬───┬───┐"
310  PRINT "│   │ 一 │ 二 │ 三 │"
320  GOSUB 400
330  FOR I = 1 TO 4
340  PRINT "│";I;" ";"│";
350  FOR J = 1 TO 3
360  PRINT A(I,J);" │";
370  NEXT J
380  PRINT
390  IF I = 4 THEN  PRINT "└───┴───┴───
┴───┘": GOTO 394
392  GOSUB 400
394  NEXT I
396  END
400  PRINT "├───┼───┼───┼───┤"
410  RETURN
```

```
]RUN
销售计算机情况

110  120  130
140  150  160
200  210  220
230  240  250

计算机价格

31000
28600
1000

销售总营业额

T(1)=6972000
T(2)=8790000
T(3)=12426000
T(4)=14244000

公 司 总 额 S=42432000

?1
1 月 份 销 售 总 额  6972000
?2
2 月 份 销 售 总 额  8790000
?3
3 月 份 销 售 总 额  12426000
?4
4 月 份 销 售 总 额  14244000
?0
S=42432000
```

	一	二	三
1	110	120	130
2	140	150	160
3	200	210	220
4	230	240	250

2. 简易售货程序

这是一个简易而实用的售货程序。它可以自动计算顾客购买某一种商品及数量的金额；也可以计算买多种商品和数量的总金额。并且可以正确结算：如果顾客钱付少了，计算机会告诉营业员，通知顾客重新付钱；反之，如果顾客钱付多了，计算机会告诉营业员应找还顾客的钱，这样，营业员既不要算帐，也避免了找钱可能出现的差错。这对减轻营业员劳动强度，提高工作效率，减少帐目差误，维护商店信誉都是有好处的。

变量说明：

E$（I）中存贮货物名称。

C（I）中存贮货物单价。

DATA区中按货物名称及货物单价顺序存放。

N$是购物名称。

N是购物数量。

S是购物总共金额。

M是顾客应付的钱。

Q是找还顾客的钱。

设计思想：

本程序共分三段，20—70句为第一段，读存商品名称和单价，80—160句为第二段，计算出售商品种类、数量和金额，170—200句为第三段，找退顾客的金额。

技巧安排：

本程序共有4处技巧：

（1）在110句中安排了一个判断语句，如果顾客购买的货物品称N$与商店中存贮的货物名称E$（I）一致，则让I赋给K，这样，在130句中就可以用C(K)*N计算第I个商品的金额。

（2）也是在110句中，当选中某种商品后，让10赋给I（本例以10种商品为例），这样，在执行120句时，I自动加1，变成11，从而立即跳出循环，减少了循环的次数，提高了程序的执行速度。

（3）180句的安排基于两个方面考虑，加0.001处理，是为了保证计算的精度，避免差错；用取整函数INT乘100和再除100，是为了金额计算准确到分。

（4）145句安排了C(K)=0，原因是防止顾客购买的货物商店没有或者营业员在输入货物名称错误时，计算错误。读者可以自行试验，去掉145句将是什么情形。

```
]LIST

10  REM H54
15  REM  简易售货程序:E$(10)存放货物名称,C(I
```

)存放货物价格

```
17  CLEAR
20  DIM E$(10),C(10)
30  FOR I = 1 TO 10
40  READ E$(I),C(I)
45  PRINT E$(I);"  ";C(I)
50  NEXT I
60  DATA BOOK,3.24,INK,0.47,DASK,42.37,PEN,2
.34,WATCH,210
70  DATA PIQIU,0.46,NIWAWA,1.35,KUZI,4.87,YI
SHANG,5.78,LIZI,1.35
80  INPUT "输入购买货物名称 N$ 和数量 N:";N$
,N
90  IF N$ = "0" OR N = 0 THEN 155
100  FOR I = 1 TO 10
110  IF N$ = E$(I) THEN  LET K = I:I = 10
120  NEXT I
130  PRINT "货物名称 ";N$;" ";"购买数量 ";N;
" ";"金额$=";C(K) * N
135  PRINT
140 S = S + C(K) * N
145 C(K) = 0
150  GOTO 80
155  PRINT
160  PRINT "总共金额:S=";S
170  INPUT "顾客应付的钱:M=";M
180 Q =  INT ((M - S) * 100 + 0.001) / 100
185  REM  顾客付钱不足转向170再重付钱
190  IF Q < 0 THEN  GOTO 170
200  PRINT "找还顾客的钱: Q=";Q
210  END
```

运行实例 1：

```
]RUN
BOOK  3.24
INK  .47
DASK  42.37
PEN  2.34
WATCH  210
PIQIU  .46
NIWAWA  1.35
KUZI  4.87
YISHANG  5.78
LIZI  1.35
输入购买货物名称 N$ 和数量 N:INK,10
货物名称 INK 购买数量 10 金额$=4.7

输入购买货物名称 N$ 和数量 N:BOOK,5
货物名称 BOOK 购买数量 5 金额$=16.2

输入购买货物名称 N$ 和数量 N:LIZI,12
货物名称 LIZI 购买数量 12 金额$=16.2

输入购买货物名称 N$ 和数量 N:0,0

总共金额:S=37.1
顾客应付的钱:M=32
顾客应付的钱:M=38
找还顾客的钱: Q=.9
```

实例 1 中，顾客购货总共金额37.1元，若顾客只付32元，则计算机会通知营业员叫顾客重付，当顾客重新付款38元时，计算机会正确找还顾客0.9元。

运行实例 2：

```
]RUN
BOOK  3.24
INK  .47
DASK  42.37
PEN  2.34
WATCH  210
PIQIU  .46
NIWAWA  1.35
KUZI  4.87
YISHANG  5.78
LIZI  1.35
输入购买货物名称 N$ 和数量 N:BOOK,20
货物名称 BOOK 购买数量 20 金额$=64.8

输入购买货物名称 N$ 和数量 N:DASK,2
货物名称 DASK 购买数量 2 金额$=84.74

输入购买货物名称 N$ 和数量 N:HHJJ,30
货物名称 HHJJ 购买数量 30 金额$=0

输入购买货物名称 N$ 和数量 N:0,0

总共金额:S=149.54
顾客应付的钱:M=150
找还顾客的钱: Q=.46
```

实例 2 中，输入 HHJJ 货物名称，这是数据区中所没有的，即使输入进去，也不会计算金额，从而避免了不应有的差错，如果没有145句 C(I)= 0 的安排，则将出现面貌全非的情况。

3. 支票查找统计程序

假设有以下支票（或存折）数据：

支票号	月一日一年	金额
1025	1.0182	10.98
1027	4.0182	39.35
3723	4.1283	95.56
4987	4.1283	80.47
8301	7.0384	50.83
9741	9.1185	60.32

试编一个BASIC程序，要求：

（1）统计所有支票金额总数；

（2）输入日期，能查找该日期支票，若相同日期有多张支票，也能正确查找；

（3）输入支票号码，查找该号码支票情况；

（4）输入日期和金额，查找同时满足日期和金额的支票信息，从而区分同一日期不同金额的支票情况；

（5）列表打印全部支票信息。

分析：由以上要求可知，这是要求编制一个多功能检索、统计、列表的程序。

为了完成上述程序设计要求，我们先画出一个粗略框图，见图5.1。

对照图5.1，我们分段写出相应程序：

由于支票共有6张，每张有3个信息，所以选择二维数组C（I,J）比较方便，C（I,1）存贮支票号码，C（I,2）存贮日期，以MMDDYY形式记录，其中MM代表月份，

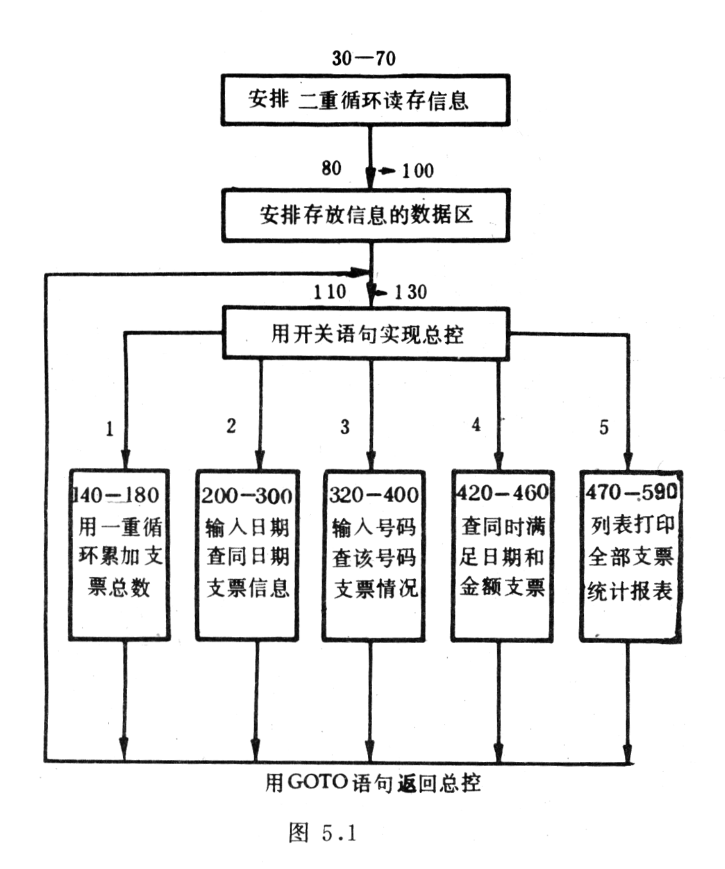


图 5.1

DD 代表日期，YY为年，如4.1283表示1983年4月12日，C(I,3) 存贮金额，I为第I张支票(本例I=1,2,3,……,6)。而对二维数组记录的读取，用二重循环最为合适，故读取信息的程序见H55:

```
28  REM H55
30  DIM C(6,3)
40  FOR I = 1 TO 6
50  FOR J = 1 TO 3
60  READ C(I,J)
70  NEXT : NEXT
80  DATA    1025,1.0182,10.98,1027,4.0182,39.35
90  DATA  3723,4.1283,95.56,4987,4.1283,80.47
100  DATA  8301,7.0384,50.83,9741,9.1185,60.32
```

程序H55中80—100句DATA语句，是放的支票信息，按支票号、日期和金额次序顺序存放。所有NEXT后面的循环变量可以省去，以利提高执行速度，中华学习机，APPLE-Ⅱ，紫金-Ⅱ以及IBM PC/XT机都有此种功能，但PC-1500机不可以省去NEXT后面的循环变量。

用开关控制转向语句ON〈算术表达式〉GOTO〈第一个语句标号〉,……〈第N个语句标号〉,来处理整个程序的连接，实现总控。

这是由于本程序设计要求能实现多功能检索，用上述开关语句，可以根据输入的不同要求，方便而又灵活地实现不同项目的查找，总控只有三行：

```
110  INPUT "#";X
120  ON X GOTO 140,200,320,420,470
130  END
```

例如，输入X＝1时，执行120句转向140句；输入X＝2时，转向200句；……；输入X＝5时，转向470句；而当X＞＝6时，相当于120句不存在，而执行120句的下一句，

即130句结束。

本程序设计的第一个要求是累加所有支票金额的总数，这个工作可由140—180句来完成，即由循环语句来实现。在循环体外赋初值S＝0，否则重新运行程序将发生错误，循环体中求和，用S＝S＋C(I，3)实现，因为C(I，3)是指第I张支票中第3个纪录的值，该值已经在第一段程序中存贮好，系指支票金额，I从1到6循环，从而完成所有支票金额的累加，循环完成打印出所有支票的金额总数，见程序H56：

```
138  REM H56
140 S = 0
150  FOR I = 1 TO 6
160 S = S + C(I,3)
170  NEXT I
180  PRINT "总数";S
190  GOTO 110
```

程序H56，在完成第一个功能打印金额总数S后，执行190句再转向110句，返回总控，从而可以再从键盘输入X值，而灵活转向其它功能的查询。可以设想，其它各功能程序段，在完成各自的功能后，一定要安排××× GOTO 110语句，以便返回总控，实现整个程序的连接和控制。

本程序的第二个要求是输入关键字日期，应能查找出该日期的所有支票信息，此项任务由200—310句完成，请看程序H57：

```
198  REM H57
200  INPUT "输入日期(MM.DDYY)";D
```

```
210  FOR I = 1 TO 6
220  IF C(I,2) = D THEN  GOSUB 240
230  GOTO 290
240  FOR J = 1 TO 3
250  PRINT C(I,J);" ";
260  NEXT J
270  PRINT
280  RETURN
290  NEXT I
300  PRINT
310  GOTO 110
```

程序H57稍许复杂一点，但还是能看懂的，现在，让我们来代替计算机运行一下该程序段。

现在假设程序已返回总控 110 句，从键盘输入 2，则转向 200 句，提示告诉我们输入日期，如此时要查找1983年 4月12日的支票，即键入4.1283，进入 210 句循环，I = 1，C(I，2) = C(1，2) = 1.0182，它和输入的值D = 4.1283不相等，执行230句转向260句加步长I = 2，再循环上去，此时C(I，2) = C(2,2) = 4.0182，又不等于D，再循环过去，I = 3，则C(I，2) = C(3，2) = 4.1283，此值与输入的D值相等，转向 240 句，注意此时I = 3，执行240—260 句一个小循环后，即将打印出第三行，1,2，3 三列的信息，即打印第三张支票的全部信息：

3723　　4.1283　　95.56

执行270句空一行，执行280句自动返回到 230 句再转向290句，加步长，I = 4，进入220句，此时C =(I，2) =C(4，2) = 4.1283，它也等于输入的D值，同上分析又打印出第 4

张支票C(4,1),C(4,2),C(4,3)的信息，即：

4987　　4.1283　　80.47

以后的执行情况，同上分析，显然，继续循环查遍所有余下来的纪录,再也找不到日期为4.1283的信息，脱离290句循环后，空一行，执行 310 句返回总控，等待其它信息的查找。

总之,本程序段功能是能找出相同日期的所有支票信息，若某一日期只有一张支票，就找出该张信息。

本程序段有两个地方值得提一下，一是240—260句一段小循环,这种安排是一种技巧，因为当I 一定时,只要进行 J 循环,即能打印I 行J 列的信息(J＝1,2,3)，而省去用I 、J 两重循环的麻烦。二是在循环中使用了转子语句GOSUB / RETURN，这是比较合适的。因为，要想在一个程序的不同地方多次使用同一程序段，而且要求用完以后返回原处的话，用转子语句是十分方便的，这样，程序写得可以更加简洁、节省，因此，这里的写法也是一个技巧。

现在，我们来看本题的第三个要求，它在程序上是如何实现的，请看程序H58中的320—410句：

```
318 REM H58
320 INPUT "敲入支票号(NUMBER): ";N
330 FOR I = 1 TO 6
340 IF C(I,1) = N THEN 370
350 NEXT I
360 PRINT "WITHOUT": GOTO 110
370 FOR J = 1 TO 3
380 PRINT C(I,J);" ";
390 NEXT J
```

```
400  PRINT
410  GOTO 110
```

这段程序比较简单，输入支票号码N后，进入330—350句循环中进行查找，若满足340句条件即找到相应的支票号转向370句开始的另一个循环，打印该支票号码的信息；若找不到则在第一个循环完即打印“没有”的信息(WITHOUT)，然后再返回总控。

从技巧上看，330—350句安排也是恰当的，因为一经查找到，立即跳出循环，从而减少了无效循环的次数，这对大量信息的查找更显出其优越性。另外，本程序段不需要像第二个程序段那样，安排GOSUB/RETURN语句，因为支票号码是各不相同的，找到某个号码只有一种可能，一经找到打印后立即返回总控。

本题的第四个要求，容易由程序段H59来实现，见420—460句：

```
418  REM H59
420  INPUT "敲入日期和金额(MM.DDYYAND MARK): ";D
,M
430  FOR I = 1 TO 6
440  IF C(I,2) = D AND C(I,3) = M THEN 370
450  NEXT I
460  PRINT "NOT  FOUND!": GOTO 110
```

本程序段完成对日期和金额都满足要求的支票进行搜索，这主要由440句的条件语句来实现，只有日期和金额两个条件都符合时，才算查找到，否则循环再找。而一经找到则转向370句开始的另一个循环，那是一段打印程序，这里处理

也是好的，它可以省去重复安排打印程序段的处理，程序比较简炼。如果两个条件有一个不符合或全不符合，则由460句出示“找不到”信息（NOT FOUND!）。

440句的写法也是一个技巧，否则要用二个条件语句处理。

本题最后一个要求，是打印报表，由程序H60完成。

```
468  REM H60
470  PRINT  TAB( 10);"** 报 表 **"
480  PRINT
490  GOSUB 600
500  PRINT "! 支票号!!  日  期 !!  金 额 !"
510  GOSUB 600
520  FOR I = 1 TO 6
530  FOR J = 1 TO 3
540  PRINT "!  ";C(I,J);" ";"!";
550  NEXT J
560  PRINT
570  GOSUB 600
580  NEXT I
590  GOTO 110
600  PRINT "------------------------------"
610  RETURN
```

本程序段阅读不会有太多的困难，470句打印一个表头，490，510，570句都是调子程序，而子程序安排在600句，它仅为一句画虚线，画好后立即返回到调它的那一语句的下一句。520—580句两重循环，是打印I行，J列的全部支票信

息。

本程序段注意表格打印技巧上的安排，主要是合理地安排了GOSUB/RETURN语句，虽然子程序十分简单，但可以通过本程序段的学习，掌握一种简便有效的打印表格方法，这一点比用作图语句绘制表格要方便灵活。

最后，我们再看一下本程序的汉字提示，它主要由20句以及620—680句完成，见程序H61：

```
]LIST

18  REM H61
20  GOSUB 620
620  PRINT "输入1:统计金额总数"
630  PRINT "输入2:敲入日期(MM.DDYY)找该日期支票"
640  PRINT "输入3:敲入支票号(NUMBER)找该支票"
650  PRINT "输入4:敲入日期與金额,找同时满足日期
與金额的支票"
660  PRINT "输入5:打印报表"
670  PRINT "输入6:结束"
680  RETURN
```

程序这样安排有几个好处：

·它是本题的菜单提示，实际上是告诉使用者本程序有五个功能。

·它也是本程序操作说明，例如，输入1统计金额总数，输入6结束。

本程序中多处使用了GOSUB/RETURN语句，我们可以概括一下它在程序设计中的要领：

(1) 在主程序中应当恰当安排调用点，作好调用准备，

以及调用返回后的必要处理；

（2）对子程序则应充分注意到其逻辑上的完整性和程序的通用性。

按照这些要领，读者仔细分析一下本程序中几段调用子程序的地方，特别是200—310句的程序段，也许是有益的。

现在，我们可以将本题的解题程序完整地写出来了，见H62：

```
5  REM H62
10  REM 支票查找统计程序
20  GOSUB 620
30  DIM C(6,3)
40  FOR I = 1 TO 6
50  FOR J = 1 TO 3
60  READ C(I,J)
70  NEXT : NEXT
80  DATA    1025,1.0182,10.98,1027,4.0182,39.35
90  DATA  3723,4.1283,95.56,4987,4.1283,80.47
100  DATA  8301,7.0384,50.83,9741,9.1185,60.32
110  INPUT "#";X
120  ON X GOTO 140,200,320,420,470
130  END
140 S = 0
150  FOR I = 1 TO 6
160 S = S + C(I,3)
170  NEXT I
180  PRINT "总数: ";S
190  GOTO 110
```

```
200  INPUT "输入日期(MM.DDYY): ";D
210  FOR I = 1 TO 6
220  IF C(I,2) = D THEN  GOSUB 240
230  GOTO 290
240  FOR J = 1 TO 3
250  PRINT C(I,J);" ";
260  NEXT J
270  PRINT
280  RETURN
290  NEXT I
300  PRINT
310  GOTO 110
320  INPUT "敲入支票号(NUMBER): ";N
330  FOR I = 1 TO 6
340  IF C(I,1) = N THEN 370
350  NEXT I
360  PRINT "WITHOUT": GOTO 110
370  FOR J = 1 TO 3
380  PRINT C(I,J);" ";
390  NEXT J
400  PRINT
410  GOTO 110
420  INPUT "敲入日期和金额(MM.DDYYAND MARK): ";D
,M
430  FOR I = 1 TO 6
440  IF C(I,2) = D AND C(I,3) = M THEN 370
450  NEXT I
460  PRINT "NOT  FOUND!": GOTO 110
470  PRINT  TAB( 10);"** 报 表 **"
480  PRINT
490  GOSUB 600
```

```
500  PRINT "! 支票号!!  日 期 !!  金 额 !"
510  GOSUB 600
520  FOR I = 1 TO 6
530  FOR J = 1 TO 3
540  PRINT "!  ";C(I,J);" ";"!";
550  NEXT J
560  PRINT
570  GOSUB 600
580  NEXT I
590  GOTO 110
600  PRINT "------------------------------"
610  RETURN
620  PRINT "输入1:统计金额总数"
630  PRINT "输入2:敲入日期(MM.DDYY)找该日期支票"
640  PRINT "输入3:敲入支票号(NUMBER)找该支票"
650  PRINT "输入4:敲入日期與金额,找同时满足日期
與金额的支票"
660  PRINT "输入5:打印报表"
670  PRINT "输入6:结束"
680  RETURN
```

```
]RUN
输入1:统计金额总数
输入2:敲入日期(MM.DDYY)找该日期支票
输入3:敲入支票号(NUMBER)找该支票
输入4:敲入日期與金额,找同时满足日期與金额的支票
输入5:打印报表
```

```
输入6:结束
#1
总数: 337.51
#2
输入日期(MM.DDYY): 4.0182
1027 4.0182 39.35

#2
输入日期(MM.DDYY): 4.1283
3723 4.1283 95.56
4987 4.1283 80.47

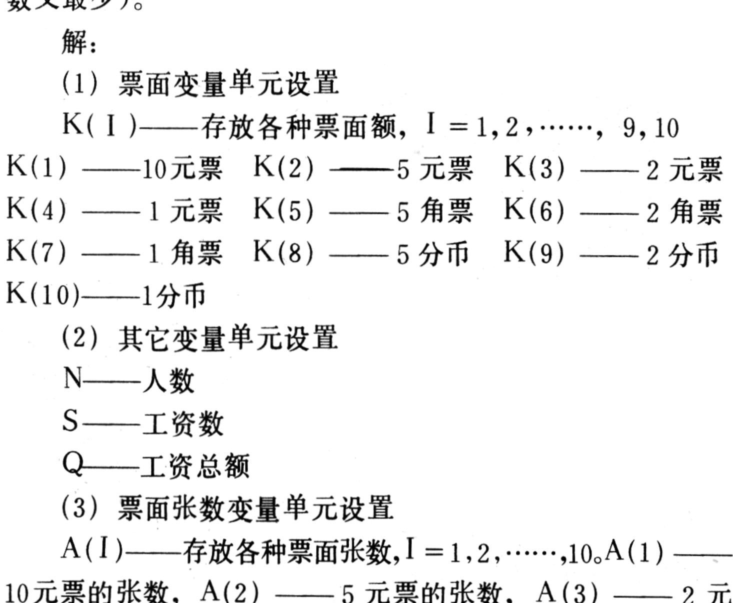
#3
敲入支票号(NUMBER): 8301
8301 7.0384 50.83
#4
敲入日期和金额(MM.DDYYAND MARK): 4.1283
??95.56
3723 4.1283 95.56
#5
          ** 报 表 **

-------------------------------
! 支票号!!  日  期 !!  金 额 !
-------------------------------
!  1025 !!  1.0182 !!  10.98 !
-------------------------------
!  1027 !!  4.0182 !!  39.35 !
-------------------------------
!  3723 !!  4.1283 !!  95.56 !
-------------------------------
```

```
! 4987 !! 4.1283 !! 80.47 !
---------------------------
! 8301 !! 7.0384 !! 50.83 !
---------------------------
! 9741 !! 9.1185 !! 60.32 !
---------------------------
```

4. 发工资程序

财务科从银行领取工资总额时，需要有各种票面的钞票才能顺利分发。试编一个程序，使财发科从银行领到的各种票面的钞票和硬币的数目最为合理（既能顺利分发，张(枚)数又最少）。



解：

(1) 票面变量单元设置

K(I)——存放各种票面额，I＝1,2，……，9,10

K(1) ——10元票　K(2) ——5 元票　K(3) ——2 元票

K(4) ——1 元票　K(5) ——5 角票　K(6) ——2 角票

K(7) ——1 角票　K(8) ——5 分币　K(9) ——2 分币

K(10)——1分币

(2) 其它变量单元设置

N——人数

S——工资数

Q——工资总额

(3) 票面张数变量单元设置

A(I)——存放各种票面张数,I＝1,2,……,10。A(1) ——10元票的张数，A(2) ——5 元票的张数，A(3) ——2 元

票的张数，……，A(9) —— 2分硬币的枚数，A(10)——1分硬币的枚数。

(4) 循环变量控制

J变量控制读取工资数

I变量控制：读各种票面值，处理表面张数，打印各种票面值。

(5) 主要程序段设计思想

将工资数分成各种票面数，是本程序设计中主要考虑的问题。

方法是安排一重循环，如：

```
┌ 380 FOR  I=1 TO 10
│     循环体
└ 420 NEXT  I
```

在循环外读工资数S，如56.76元。

循环体中安排以下语句：

```
┌──→ M=INT(S/K(I))
│           ↓
│    S=S-M*K(I)
│           ↓
│    A(I)=A(I)+M
└───────────↓
```

第一次：I=1，K(1)=10，M=5，S=56.76－5×10=6.76。

A(1)=A(1)+M=0+5=5，表明10元钱的5张。

第二次：I=2，K(2)=5，M=1，S=6.76－1×5=

1.76。

A(2)=A(2)+M=0+1=1,即5元钱的1张。

第三次：I=3,K(3)=2,M=0, S=1.76−0×2=1.76。

A(3)=A(3)+M=0+0=0，即没有2元钱。

第四次：I=4,K(4)=1,M=1, S=1.76−1×1=0.76。

A(4)=A(4)+M=0+1=1,即1元的1张。

第五次：I=5,K(5)=0.5，M=1，S=0.76−0.5=0.26。

A(5)=A(5)+M=0+1=1,即5角的1张。

第六次：I=6,K(6)=0.2，M=1，S=0.26−1×0.2=0.06。

A(6)=A(6)+M=0+1=1，即2角的一张。

以下过程同上分析，直到将工资数分成最小票面数1分硬币为止，I循环结束，再重新处理第2个人的工资情况。

十分明显，各种票面应首先读入存贮，请见程序H63中310—340句循环。

其它几个问题的处理比较简单，例如读工资数，累加，做必要的精度处理，打印工资总额，以及打印各种票面的张数等。我们在程序中都附加了中文说明，很容易阅读和理

解。

下面给出 5 个人的运行结果:

```
]RUN
N=5
Q=290.96
10:27
5:1
2:4
1:5
.5:4
.2:3
.1:2
.05:2
.02:2
.01:2
```

源程序:

```
]LIST

2  REM H63
5  REM 发工资程序
10  INPUT "N=";N: REM  输入人数
20  GOSUB 300
30  END
300  DIM K(10),A(10)
310  FOR I = 1 TO 10
320  READ K(I): REM 读入各种票面
330  DATA 10,5,2,1,.5,.2,.1,.05,.02,.01
```

```
340  NEXT I
350  FOR J = 1 TO N
360  READ S: REM 工资数
365 S = S + 0.001: REM 精度处理
370 Q = Q + S: REM 累加工资数
375  REM 380-42将工资数分成各种票面数
380  FOR I = 1 TO 10
390 M =  INT (S / K(I))
400 S = S - M * K(I): REM 工资余数
410 A(I) = A(I) + M: REM 票面张数
420  NEXT I
430  NEXT J
440  PRINT "Q=";Q - 0.001 * N
450  FOR I = 1 TO 10
460  PRINT K(I);":";
470  PRINT A(I)
480  NEXT I
490  RETURN
500  DATA 56.76,43.19,33.91
510  DATA 83.5,73.6,150,213.78
520  DATA 67.98,100,32.45,123.78
```

5. 工资程序

某单位要求用中华学习机计算并打印出：

· 职工月工资表

· 各工资项合计

同时要求各主要程序段应有简要的中文说明，报表头及姓名也用汉语填写。

分析：

（1）首先确定工资表格式及项目

为简单计以5个职工为例，每个职工共有编号、姓名、基本工资、工龄补贴、物价补贴、洗理费、应发工资、会费、房租、储蓄、其它及实发工资等12项。形式如下：

编号	姓名	基资	龄贴	物补	洗理	应发	会费	房租	储蓄	其它	实发
1001	周光来	113	18	8.25	4.17	143.42	.65	2.75	5.78	2.65	131.59
1002	郑　生	105	12.5	8.25	4.17	129.92	.55	5.75	8.78	1.35	113.49
1003	申　剑	131	19.5	8.25	4.17	162.92	2.32	8.65	9.78	4.35	137.82
1004	秦　松	105	12.5	8.25	4.17	129.92	3.32	4.65	2.78	4.15	115.02
1005	马立文	122	10.5	8.25	4.17	144.92	1.12	2.25	2.28	4.25	135.02
合计		576	73	41.25	20.85	711.1	7.96	24.05	29.4	16.75	632.94

（2）其次确定变量安排

姓名用字符串变量A＄(I)存贮，I＝1为第一个职工姓名，I＝2为第二个职工姓名，其余依次类推。程序开头应确定A＄(I)中的I值，如本单位有200名职工，则应有 DIM A＄(200) 语句。

对于每一个职工的各工资项目，因为都是数值型的，不需要安排字符型变量，而采用数值型的变量，用数组A(I，J)比较方便。A(I，J)表示第I个职工第J个项目的存贮信息。例如A(2，2)，表示第2个职工(本题是张小民)，第2项（本题是工龄补贴(2.45元）的信息。对于A(I，J)数组的说明，同样应在程序开始出现，例如A(1，2)，表示A(0，0)，A(0，1)，A(0，2)，A(1，0)，A(1，1)，A(1，2)共6个存贮单元。数组说明语句DIM　A(I,J)中的I，J值，不能

太大，也不能太小，太大占用内存多，太少不够存贮，本题中存贮单元略大于各工资项。

DATA语句中存放的是工资信息，每一条DATA语句，存放一个职工信息，存放次序应有严格规定，各项之间应用“，”分隔。如一个叫李民的人，他的工资情况放在330句的DATA语句中，按姓名（李民），基资（113.55），龄贴(18.00)，物补(8.25)，洗理(4.17)，会费(0.65)，房租(2.75)，储蓄（5.78)，其它（2.65）依次存放。

各工资项存贮情况如下：

A(I，1)——基资

A(I，2)——龄贴

A(I，3)——物补

A(I，4)——洗理

A(I，6)——会费

A(I，7)——房租

A(I，8)——储蓄

A(I，9)——其它

对于每个职工的各项收入，放在A(I，5)存贮单元中，它实际上是每个职工应发工资总额，包括基资，龄贴，物补，洗理四项收入的累加；而每个职工的各项支出，放在A(I，11)存贮单元中，它应是每个职工扣除的所有支出部份，包括会费，房租，储蓄和其它四项；而每个职工收入A(I，5)减去支出A(I，11)放在A(I，10)存贮单元中，这就是每个职工实发工资。

对于每一工资项的累加和合计，放在A(0，I)中。

(3) 各程序段功能及简要说明

第一段：50—160句读入工资信息。本程序段共有三重循环组成，最外面一个循环用变量I控制，这是处理每一个人工资情况。中间用两个循环，70句—100句读收入，全部收入放在A(I，5)中；110句—140句读支出，全部支出放在A(I，11)中。在内循环完成以后，计算每个职工实发工资数，并放在A(I，10)中。以后再循环上去依次读其它各职工工资信息，并完成同上的处理工作。本段是整个程序的核心部份。

第二段：210—260句是打印定位，用一重循环完成，其中，I作为循环控变量，又兼各人序号安排。所有打印位置均用TAB(X)语句。这一段程序执行后，完成全体职工各项工资信息的打印输出。

第三段：270—300句，是合计汇总，完成各工资项的累加。采用调子程序的方法处理，执行270句时，调380句开始的一段子程序，子程序放在380—420句的两重循环中，它们处理各工资项累加，完成后通过430句RETURN语句返回，再经过280—300句定位打印，最后结束停机。

(4) 几个语句的安排说明

·30句安排了一个防止出错语句，这是由于本题为说明原理简单起见，仅在DATA语句中存放了5个人的信息，而50句循环终值为15，这样READ/DATA不配对，安排了30句出错自动处理以后，程序能正确运行。实行使用本程序时，可分别两种情况处理：一是删去30句，将I的终值定为和存贮人数一致的数值；二是继续保留30句，将50句I的终值再放大一点，例如I=250,而DATA语句中人员数量可以任意增多（不超过250）或减少，从而使程序更加灵活和

通用，使程序增加了增加数据和减少数据的功能。

·170句是程序自动接通打印机，并设置了第一种字型的打印格式，当然，如实行打印要求时，应在程序运行前接通打印机电源。改变POKE 1659，N中的N值（N取1—5为宜）改变字体大小。310句POKE 1659，0 是断开打印机指令。

·180句中POKE 2043，60，是设置每行字数最多可以打印60个字符，如不安排这一句，计算机按40个字符处理，本表打印不能对齐。

·本程序中用了不少REM语句，目的是对程序加注了必要的中文注释说明，这样做有几个好处，程序易读性好，又能看清程序的执行过程。同时，程序不论放置多少时间，再重新阅读也非常方便。

·请注意本程序中TAB（X）语句的用法，中华学习机在中文状态下，每行只处理17个汉字，34个字符，若超过34个字符时用TAB函数会出错，方法是第一个超过34的保持，如TAB（35）是可以的，但以后必须减去34。例如要在43号位置上打印，不能写TAB（43），而改写TAB（9），其中43－34＝9。即以34为模。

程序清单（见H64）和运行结果：

```
]LIST,152

5  REM H64
10  REM 工资程序
20  DIM A(15,11),A$(15)
```

```
25 ONERR GOTO 150
28 REM 30-140读入工资信息
30 FOR I = 1 TO 15
40 READ A$(I): REM 读姓名
50 FOR J = 1 TO 4
60 READ A(I,J): REM 读收入
70 A(I,5) = A(I,5) + A(I,J): REM 应发
80 NEXT J
90 FOR J = 6 TO 9
100 READ A(I,J): REM 读支出
110 A(I,11) = A(I,11) + A(I,J): REM 支出相加
120 NEXT J
130 A(I,10) = A(I,5) - A(I,11): REM 收入减支出
140 NEXT I
150 POKE 1659,5
152 POKE 2043,60: REM 设置字数
155 PRINT "编号 姓 名基 资龄 贴物 补洗
 理应 发会 费房 租储 蓄其 它实 发"
158 PRINT : REM 160-200打印定位
160 FOR I = 1 TO 5
170 PRINT I + 1000;" ";A$(I);" ";
180 PRINT TAB( 13);A(I,1); TAB( 21);A(I,2); TA
B( 28);A(I,3); TAB( 35);A(I,4); TAB( 7);A(I,5);
185 PRINT TAB( 16);A(I,6); TAB( 22);A(I,7); TA
B( 29);A(I,8); TAB( 36);A(I,9); TAB( 8);A(I,10)
190 PRINT
200 NEXT I
205 GOSUB 500: REM 合计汇总
210 PRINT "   合  计";
220 PRINT TAB( 13);A(0,1); TAB( 21);A(0,2); TA
```

```
B( 28);A(0,3); TAB( 35);A(0,4); TAB( 7);A(0,5);
230  PRINT  TAB( 16);A(0,6); TAB( 22);A(0,7); TA
B( 29);A(0,8); TAB( 36);A(0,9); TAB( 8);A(0,10)
240  POKE 1659,0
250  END
400  DATA 周光来,113.00,18.00,8.25,4.17,0.65,2.7
5,5.78,2.65
410  DATA 郑  生,105.00,12.50,8.25,4.17,0.55,5.7
5,8.78,1.35
420  DATA 申  剑,131.00,19.50,8.25,4.17,2.32,8.6
5,9.78,4.35
430  DATA 秦  松,105.00,12.50,8.25,4.17,3.32,4.6
5,2.78,4.15
440  DATA 马立文,122.00,10.50,8.25,4.17,1.12,2.2
5,2.28,4.25
500  FOR I = 1 TO 10
510  FOR J = 1 TO 5
520 A(0,I) = A(0,I) + A(J,I)
530  NEXT J
540  NEXT I
550  RETURN
```

6. 图书管理程序

(1) 程序简介

本程序利用字符串比较这一特性，在DATA语句中放进全部书的编号、书名、作者姓名和出版年月，如果想查询有没有某一本书，可以从键盘输入此书名字，如有此书，

编号	姓名	基资	龄贴	物补	洗理	应发	会费	房租	储蓄	其它	实发
1001	周光来	113	18	8.25	4.17	143.42	.65	2.75	5.78	2.65	131.59
1002	郑 生	105	12.5	8.25	4.17	129.92	.55	5.75	8.78	1.35	113.49
1003	申 剑	131	19.5	8.25	4.17	162.92	2.32	8.65	9.78	4.35	137.82
1004	黎 松	105	12.5	8.25	4.17	129.92	3.32	4.65	2.78	4.15	115.02
1005	马立文	122	10.5	8.25	4.17	144.92	1.12	2.25	2.28	4.25	135.02
合计		576	73	41.25	20.85	711.1	7.96	24.05	29.4	16.75	632.94

则会打印出此书的书号、书名、作者和出版时间。如找不到此书，则打印“对不起！找不到”的信息。如果还想继续查找其它书，程序中安排了控制语句，只要按Y键↙便可重新查找；如果不想再找，按N键↙停止。操作简单，使用方便，适用于图书资料管理及情报检索中使用。

（2）功能特点

· 本程序有两种查询方式，一是根据书名查找；二是根据作者姓名和书名两个条件查找。对于后者，由于书名可能相同，但作者一般不会相同，这样，就可以正确查询同时满足书名和作者两个条件的书，而对前者，则可以查找相同书名的书。

· 本程序既可以查找英文书，也可以查找中文书，而不需要为了区别英文和中文另编程序，对小型图书馆更为实用灵便。

（3）变量安排

· N——书的编号

· M——图书馆中图书总数

· B $——书名

· C $——作者姓名

· X——出版年月

· A $——查询的书名（仅按书名查找）

· D $——查询的作者姓名

（4）操作说明

步骤	输　入	显　　示	注　释
1	RUN ↙	ARE THERE BOOKS IN YOUR LIBRARY?	图书馆里有多少书?
2	8 ↙	TYPE BOOK NAME?	输入8,并指示打入书名
3	FORTRAN-77 ↙	1129 FORTRAN-77 SMITH 1983.6	输入书名，显示出查找到的该书信息
		TO CONTINUE TYPEY; ELES N	提示是否查找，是按Y，否则按N
4	Y ↙	作者: D$ 书名: A$	输入Y，将提示打入作者和书名
5	TIME ↙	? ?	输入作者，提示书名
6	BASIC↙	2741 BASIC TIME 1976. 12]	输入书名，打印该书信息 结束

(5) 几个运行实例

例 1：

```
]RUN
----------------------------------------
HOW MANY BOOKS ARE THERE IN YOUR LIBRARY?8
TYPE BOOK NAME? FORTRAN-77
1129        FORTRAN-77  SMITH       1983.6
TO CONTIUNE TYPE Y;ELES N Y
作者:D$ 书名:A$ TIME
??BASIC
2741        BASIC       TIME        1976.12
```

例 2：

```
]RUN
----------------------------------------
HOW MANY BOOKS ARE THERE IN YOUR LIBRARY? 8
TYPE BOOK NAME? 王林
对不起!找不到
TO CONTIUNE TYPE Y;ELES N Y
作者:D$ 书名:A$ 朱国江
??中华学习机编程技巧
3424          中华学习机编程技巧朱国江
                         1988.2
```

书名输入错误，找不到。

例 3：

```
]RUN
----------------------------------------
HOW MANY BOOKS ARE THERE IN YOUR LIBRARY?8
TYPE BOOK NAME? APPLE-2
对不起!找不到
TO CONTIUNE TYPE Y;ELES N N
```

中文书名必需在中文状态下输入。

(6) 程序清单

```
]LIST,110

5  REM H65
10  REM **FIND BOOK NAME NUMBER AND WRITER PROGR
AM**
```

```
15  REM  查找编号.书名.作者和出版年月
17  REM TAB( 1);"编号"; TAB( 13);"书名"; TAB( 25
);"作者"; TAB( 37);"出版年月"
18  FOR K = 1 TO 40: PRINT "-";: NEXT K: PRINT
19  REM "图书馆中共有多少本书?M本"
20  INPUT "HOW MANY BOOKS ARE THERE IN YOUR LIBR
ARY?";M
22  REM "输入查找的书名 A$"
30  INPUT "TYPE BOOK NAME?";A$
40  FOR I = 1 TO M
45  REM  编号:N 书名:B$ 作者:C$ 出版年月:X
50  READ N,B$,C$,X
60  IF B$ = A$ THEN 100
70  NEXT I
80  PRINT "对不起!找不到"
85  REM  YOU CAN NOT FIND THIS BOOK
90  GOTO 120
97  PRINT  TAB( 1);"编号"; TAB( 13);"书名"; TAB(
 25);"作者"; TAB( 37);"出版年月"
100  PRINT  TAB( 1);N; TAB( 13);B$; TAB( 25);C$;
 TAB( 37);X
110  REM  是否继续找,是按Y键,否按N键

120  INPUT "TO CONTIUNE TYPE Y;ELES N";D$
130  IF D$ < > "Y" THEN 160
140  RESTORE
150  GOTO 200
160  END
200  INPUT "作者:D$ 书名:A$ ";D$,A$
205  REM  查找同时满足两个条件的书
210  FOR I = 1 TO M
```

```
220  READ N,B$,C$,X
240  IF A$ = B$ AND D$ = C$ THEN 290
260  NEXT I
270  PRINT "------NOT FOUND-----"
280  END
290  PRINT  TAB( 1);N; TAB( 13);B$; TAB( 25);C$;
 TAB( 37);X
300  DATA 2741,BASIC,TIME,1976.12
310  DATA  3424,中华学习机编程技巧,朱国江,1988.2

320  DATA 1129, FORTRAN-77,SMITH,1983.6
330  DATA  5672,扩充BASIC,常  铭,1987.3
340  DATA  8987,EXPERT SYSTEM,孙  云,1987.8
350  DATA  4547,LISP语言,商  亮,1979.8
360  DATA 5566,人工智能,宋里,1987.
370  DATA 9912,APPLE,朱同,1979.3
```

7．人事档案管理程序

本程序以8个人员，每人7个信息为例，介绍了一种编制人事档案的原理和方法，为简单计，档案中人员姓名用三个英语字母代替，性别用M和W代替。

（1）本程序可以完成下述功能

· 列印全部人员档案；

· 按人员序号查找该员档案；

· 查找小于某个年龄的人员名单；

· 查找符合多少年龄以下，工龄多少年以上，某一文化程度三个条件的人员情况；

· 查找某一文化程度占总人数的比例；

· 查找某一性别，某种年龄以下，某一文化程度人员资料。

(2) 程序分段说明

· 1句文件名：人事档案管理程序。

· 5—12句：读存信息，将全部人员档案资料存入数组A$(I,J)中。

· 24句：列菜单，共7项，数字代表输入方式，文字是功能说明，采用调子程序的方法，子程序安排在800—900句中。

· 25—35句：总控。

· 40—90句：为调用6个子程序的指令语句，全部采用GOSUB/RETURN语句，而用GOTO语句返回总控。

· 99—125句：为第一个子程序，完成列印全部人员档案资料。

· 139—160句：为第二个子程序，按序号查找人员档案。

· 179—200句：为第三个子程序，找小于某年龄人员名单。

· 209—240句：为第四个子程序，找符合年龄，工龄及文化程度三个条件的人员情况。

· 250—300句：数据区，RESTORE为恢复数据区指针。

· 309—370句：为第五个子程序，找满足某一文化程度的人数及占总人数的百分比。

· 409—460句：为第六个子程序，查同时符合性别，年

龄及文化程度三个条件的人员情况。

(3) 程序变量安排

A$(I，J)：存放档案信息，I＝1,2,…，7，8代表I个人，J＝1,2,…，7代表J项信息

#：总控提示符。要求从键盘输入1,2,3,4,5,6,7中的任何一个字码。

LL：年龄

GL：工龄

WH：文化程度，用1,2,3区别

T$：性别，M为男同志，W为女同志

FZ：工资

S：某种文化程度人数

R：某种文化程度人数占总数百分比

I，J：循环变量，I兼作人员代号

数据区中存贮情况按以下顺序：人员代号、姓名、性别、年龄、工龄、文化程序、行政级别。

(4) 主要程序段分析

① 列印全部人员档案

程序运行后，屏幕上列出一个菜单，然后出现一个#号，此时输入1↙，即可列印全部人员档案。其执行过程如下，X＝1，经30句转向40句，调99句开始的一段子程序，用两重循环，打印存贮在A$(I，J)中的档案信息，打印完毕后经125句RETURN返回到总控25句。

② 按序号查找人员档案

键盘输入2↙，执行50句，调139句开始的子程序，然后输入一个序号(1,2,……,7，8中的一个)，用一重循环145—

150句,即可查找到某序号的人员情况，所以安排一重循环，是因为 I 一定,查找人员就定。不需要再用循环查找其它人，而只要安排一个 J 循环，即可打印第 I 个人的 J 个信息。

③ 查小于等于某一年龄的人员情况

键盘输入 3↙,执行60句,调 179 句开始的子程序，然后输入一个年龄,例如40,经 180 句时，用了一个 STR $(X)函数，这是一个将数值X赋给字符串变量的函数，经此函数转换后打印出来的是"4"，"0"两个字符。此外，在 180 句中还用了一个赋值语句 J =4,它是指向每个人员档案信息的第四项年龄，这种安排是一个技巧，它和190—200句的另一个循环配合使用,可以减少一重 I 的循环。190句的条件语句中用了一个VAL (A$(I,J))函数,它的功能是把字符串变成数值，只要满足小于或等于年龄（LL）的值即打印，其中A$(I, 2)中的 2 为每一人的第二项信息即姓名，反之再继续循环查找。查找完毕返回总控。

④ 关于其它几个程序段

读者可以自行分析。只是要注意一点，对于同时查找满足几个条件的人员情况，我们用了逻辑表达式，例如， IF VAL(A$ (I, 4) <= LL AND VAL (A$(I,5)) >GL AND VAL (A$(I, 6)) = WH THEN GOSUB 145,这句话的意思是如果第 I 个人的年龄(J=4,第四项信息为年龄)小于等于LL(例如40)同时工龄(J=5,第五项为工龄) >GL (例如10) 并且文化程度是WH(如 1 代表大学毕业)的话，就转向145句去打印。

本程序只是一种模式，它可以再行扩充和修改，可以完成更多人员存贮以及更多功能的检索。程序清单（见H66）

及运行实例：

```
]LIST,95

0  REM H66
1  REM  人事档案管理程序
2  DIM A$(8,7)
4  REM  5_12:读存信息
5  FOR I = 1 TO 8: FOR J = 1 TO 7
10  READ A$(I,J)
12  NEXT J: NEXT I
15  REM  输入方式 (1,2,3,4,5,6,7)
20  REM 编号 姓  名 性  别(T$,M,W) 工龄(GL) 文化
程度(WH) 工 资(GZ)
22  REM  25_35:总 控
24  GOSUB 800
25  INPUT "#:";X
30  ON X GOTO 40,50,60,70,80,90,94
35  GOTO 25
40  GOSUB 99: GOTO 25
50  GOSUB 139: GOTO 25
60  GOSUB 179: GOTO 25
70  GOSUB 209: GOTO 25
80  GOSUB 309: GOTO 25
90  GOSUB 409: GOTO 25
94  PRINT "停止"
95  END
99  PRINT "列印全部人员档案"
100  PRINT : FOR I = 1 TO 8
105  FOR J = 1 TO 7
110  PRINT A$(I,J);" ";
120  NEXT J: PRINT : PRINT : NEXT I
125  RETURN
```

```
139  PRINT "按序号查找人员档案"
140  PRINT : INPUT "序号";I
145  FOR J = 1 TO 7
150  PRINT A$(I,J);" ";: NEXT J
160  PRINT : RETURN
179  PRINT "查找<50岁的人员名单"
180  PRINT : INPUT "LL?";LL:J = 4: PRINT "LL=";
STR$ (LL): PRINT
190  FOR I = 1 TO 8: IF  VAL (A$(I,J)) <  = LL T
HEN  PRINT  TAB( 6);A$(I,2)
200  NEXT I: RETURN
209  PRINT "找年龄<50,工龄>10年,某一文化程度的人
"

210  INPUT "LL?";LL: PRINT  STR$ (LL): INPUT "GL
?";GL: PRINT  STR$ (GL): INPUT "WH?";WH: PRINT
STR$ (WH)
220  FOR I = 1 TO 8
230  IF  VAL (A$(I,4)) <  = LL AND  VAL (A$(I,5)
) > GL AND  VAL (A$(I,6)) = WH THEN  GOSUB 145
240  NEXT I: PRINT : RETURN
250  DATA  1,ABC,W,38,15,2,21
255  DATA  2,DEF,M,44,20,1,18
260  DATA  3,GHI,M,63,38,3,14
265  DATA  4,JKL,W,41,16,1,19
270  DATA  5,MNO,M,28,10,2,22
275  DATA  6,PQR,W,38,15,1,20
280  DATA 7,STO,M,49,28,2,16
285  DATA 8,VWX,W,55,30,3,17
300  RESTORE
309  PRINT "某一文化程度占总人数比例"
310  INPUT "WH?";WH: PRINT "WH="; STR$ (WH)
```

```
315 S = 0
320  FOR I = 1 TO 8
330  IF  VAL (A$(I,6)) = WH THEN 340
335  GOTO 350
340 S = S + 1
350  NEXT I
360  PRINT "S=";S;" ";"R=";S / 8 * 100;"%"
370  RETURN
409  PRINT "查某一性别,年龄<40,某一文化程度人员
情况"
410  INPUT "T$?";T$: PRINT "T$=";T$: INPUT "LL?"
;LL: PRINT  STR$ (LL): INPUT "WH?";WH: PRINT  ST
R$ (WH)
420  PRINT
430  FOR I = 1 TO 8
435  FOR J = 1 TO 7
437  IF T$ = A$(I,3) THEN 440
438  GOTO 445
440  IF  VAL (A$(I,4)) <  = LL AND  VAL (A$(I,6)
) = WH THEN  PRINT A$(I,J);"  ";
445  NEXT J
450  NEXT I
455  PRINT
460  RETURN
800  PRINT "1:列印全部人员档案"
810  PRINT "2:按序号查找人员档案"
820  PRINT "3:查找<50岁的人员名单"
830  PRINT "4:找年龄<50,工龄>15年,某一文化程度的
人"
840  PRINT "5:某一文化程度占总人数比例"
```

```
850  PRINT "6:查某一性别,年龄<40,某一文化程度人
员情况"
860  PRINT "7:停止"
900  RETURN
```

```
]RUN
1:列印全部人员档案
2:按序号查找人员档案
3:查找<50岁的人员名单
4:找年龄<50,工龄>15年,某一文化程度的人
5:某一文化程度占总人数比例
6:查某一性别,年龄<40,某一文化程度人员情况
7:停止
#:1
列印全部人员档案

1 ABC W 38 15 2 21

2 DEF M 44 20 1 18

3 GHI M 63 38 3 14

4 JKL W 41 16 1 19

5 MNO M 28 10 2 22

6 PQR W 38 15 1 20

7 STO M 49 28 2 16

8 VWX W 55 30 3 17
```

```
#:2
按序号查找人员档案

序号 6
6 PQR W 38 15 1 20
#:3
查找<50岁的人员名单

LL?40
LL=40

     ABC
     MNO
     PQR

#:4
找年龄<50,工龄>10年,某一文化程度的人
LL?45
45
GL?10
10
WH?2
2
1 ABC W 38 15 2 21

#:5
某一文化程度占总人数比例
WH?1
WH=1
S=3 R=37.5%
#:6
查某一性别,年龄<40,某一文化程度人员情况
T$?W
```

```
I$=W
LL?38
38
WH?1
1

6  PQR  W  38  15  1  20
#:7
停止
```

六、汉 字 原 理

1. 汉字原理程序

在我国大量的信息是中文汉字，因此中文信息处理是我国计算机系统应用和研究的一个重要领域，是使计算机在我国普及推广的前提。

计算机为什么能写汉字，它的原理和方法是什么？这里介绍一种简单的实现点阵汉字的原理和方法，只是提供一种思路，从中可以了解到计算机处理汉字信息的概略过程。

计算机处理汉字的过程，可以用程序来实现，整个程序可分为建立字库、识别查找、进制转换、汉字再现四个部份。

(1) 建立字库

字库，顾名思义，不是一个汉字，而是由许多汉字组成。存放汉字的地方就是“库”。每一个汉字又不是汉字本身，而是用一定的数字代码来代替的。我们要建立的汉字库，在程序中是放在数据区中（如DATA语句中），存放的是相应汉字的数字代码。

因此，首要的问题是如何将汉字转换成相应的汉字代码。常用的方法是所谓点阵法，又称点阵模式，计算机系统中的汉字也常用点阵字模。

例如，本程序采用12×9点阵模式，即一个汉字由108个“B”字符或空格组成。方法是先画好一个方格图，12行9列，

然后在图中写字，下面是汉字“计”的实例，见图6.1。

写字的地方用“1”表示，不写字的地方用“0”代替，就可以将任何一个汉字变成相应的代码。

如对应于汉字“计”的二进制代码（见表6.1）为。

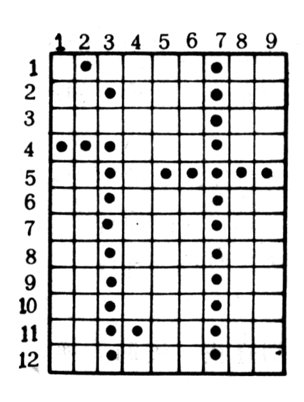


图 6.1

由此可知，采用12×9的点阵模式，处理一个汉字成二进制代码，需要108个“0”和“1”。显然，把这么多的代码存放在DATA语句中，是很不方便的，稍有误差就发生错误。

容易想到，将二进制代码转换成其它进制代码，书写就比较简短，例如转换成八进制。八进制共有0,1,2,3,4,5,6,7八个字码(又称数字代码，数码)，它的特点是逢八进一。

将二进制转换成八进制，方法很简单，每3位二进制代码用一位八进制代码表示，例如：$(010\ 110\ 111)_2=(2\ 6\ 7)_8$，它们是按8421码处理的。如二进制码101转换成八进制数，

表 6.1

	1	2	3	4	5	6	7	8	9
1	0	1	0	0	0	0	1	0	0
2	0	0	1	0	0	0	1	0	0
3	0	0	0	0	0	0	1	0	0
4	1	1	1	0	0	0	1	0	0
5	0	0	1	0	1	1	1	1	1
6	0	0	1	0	0	0	1	0	0
7	0	0	1	0	0	0	1	0	0
8	0	0	1	0	0	0	1	0	0
9	0	0	1	0	0	0	1	0	0
10	0	0	1	0	0	0	1	0	0
11	0	0	1	1	0	0	1	0	0
12	0	0	1	0	0	0	1	0	0

就是把每一位数码乘上该位的“权”就等于该位数，然后将各位数求和，就是八进制数。即：

$$1\times4+0\times2+1\times1=4+0+1=5$$

用上述方法，将汉字写成八进制代码就是：

204，104，004，704，137，104，104，104，104,104，144,104

经二到八进制转换后，汉字“计”的代码只有12组，每组3个数码共36个数字符号，不像二进制码处理共有108个数码，从而使存贮在数据区中的信息大大减少。

二进制到八进制的转换，它可以非常直观而又迅速地从方格图中一次读出来，而不需要经过编程实现。

综上所述，将汉字→写进方格→自然转换成二进制码→

再读出来→就是八进制码→顺序写进DATA区（每3位一组，中间用，号隔开）——就建立了一个汉字。将许多汉字经同样方法处理，一一写进数据区，就完成了汉字建库工作。

(2) 识别查找

对应于每个汉字共有12组数字代码，怎样识别它们而不致弄错呢？这就要给每一个汉字编个“号”，也就是说，为了找到某个汉字，首先必须先给这个汉字起个名字，“号”又称关键字或叫特征字。

例如“计”这个汉字，用一个汉语拼音（或中文字，或英语字母）的代码来代替，如JI 1，这就是关键字，将它放在数据区中对应“计”字的12组代码前面，以作识别“计”字的标志。

同样，对于其它每个汉字，都冠以一个拼音代码和12组数字代码共13组数据，分别置于DATA语句中，以备读入后“调用”。

调用的方法很简单，输入字的代码，把它放在字符串变量S $ 中，汉字特征字代码用读语句READ H$ 读取，只要H$ 和S $ 相符合，就能识别查找到。

总之，用输入的代码与汉字代码是否相符来识别。

但是，仅仅识别了要查找的汉字特征码（如计字为JI1）还不够，因为它后面还有12组数字代码（八进制）并未处理，要将这些数字代码变成汉字，必须经过“翻译”过程，才能实现汉字再现。

(3) 八一十进制转换

前面我们已经分析，为建立汉字库时的方便，在读写出

对应于汉字“黑白”点各数据时，采用了八进制数，因为它比二进制数书写起来简短明了又不容易出错。

但是计算机BASIC语言解释系统中，却是按十进制处理的，也就是说，DATA语句中存放的八进制代码，计算机误作十进制数处理，从而要发生差错，因此，还必须把八进制数的数据一个一个地转换成十进制数。

八—十进制数的转换公式为：

$$N_{10}=\sum_{i=n}^{1}a_i\times 8^{i-1}$$

$$=a_n\times 8^{n-1}+a_{n-1}\times 8^{n-2}+a_{n-2}\times 8^{n-3}+\cdots\cdots$$

$$+a_1\times 8^0$$

其中a_n，a_{n-1}，……，a_1分别为八进制数的各位数值，而8^{n-1}，8^{n-2}，……为相应 n位，n－1位，……的“权”。也就是八进制数每一位数码×该位的权等于该位的数，各位数累加就是对应于八进制数的十进制数。如：

$$(204)_8=(?)_{10}$$

$$2\times 8^2+0\times 8^1+4\times 8^0$$

$$=2\times 64+0+4\times 1$$

$$=(132)_{10}$$

$$\therefore (204)_8=(132)_{10}$$

八进制数转换成十进制数，可以用程序来实现。

为了理解上述转换过程，我们编制一个BASIC程序H67，看它是如何一位一位处理的：

```
]LIST

165  REM H67
168  DIM D(15),A(15)
170  FOR I = 1 TO 1
180  FOR J = 1 TO 3
190  READ A(J)
200  IF A(J) >  = 0 AND A(J) <  = 7 THEN 230
210  PRINT "DATA ERROR"
220  GOTO 190
230  NEXT J
240 X = 0
250  FOR J = 1 TO 3
260 X = X + 8 ^ (3 - J) * A(J)
270  NEXT J
280 D(I) = X
285  PRINT D(I)
290  NEXT I
300  DATA 2,0,4
310  END
```

这个程序比较简单，经180—230句处理后，A(1)=2，A(2)=0，A(3)=4，经250—270句处理后，X=132，最后放在D(1)中。这样就完成了八进制数到十进制数的转换，即 $(204)_8 = (132)_{10}$。如 I 的终值改为12，则上述程序即可处理12组八进制代码36位数字的转换成十进制的工作。

(4) 输出打印汉字

八—十进制数的转换已经由计算机完成，现在，是否可

以输出打印使汉字再现呢？我们说还不行，必须设法控制打印机，让它听我们指挥，在有字迹的地方，打印一个“B”字符，没有写字的部份打印一个空格，这样才能反映最初写字时对应的二进制码。即二进制的“0”，是没有字迹代表空格，二进制的“1”,是写字迹的要打印一个“B”。

计算机打印机输出时是按自左至右，由上而下逐行打印方式进行的，也就是说打印针头是自左至右，自上而下逐行滑动打印的。什么时候打印，什么时候不打印，要由程序控制。为了说明这个过程，我们不妨将汉字“计”的第一行各列情况重新列表（见表6.2)。

表 6.2

	列								
第一行	9	8	7	6	5	4	3	2	1
权	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
二进制	0	1	0	0	0	0	1	0	0
十进制		1			3			2	
八进制		2			0			4	

程序的编制要按下述规则处理：

由对应的十进制数是否大于或等于这一列所对应的权（如第9列为2^8，第8列为2^7，依次类推）而决定打印“B”或者“␣”（跳过这一列)。每打印一次“B”,十进制数应减去这一列对应的权，如此一列列处理（自左至右，从第9列直到这一行的最后一列)，而当十进制数为零时，表明这一行打印结束，继续打印下一行，直到12行全部打印完为止。

如十进制数132，从第9列开始判断，因132＜2^8（第9

列的权，$2^8=256$)，则不打印（计字第一行第 9 列二进制码为0,打印针头右移一列)。

现在打印针头指向第 8 列，对应的权为$2^7=128$，此时$132>2^7$，则应打印“B”，(计字第一行第8列对应的二进制码为1，要打印)。打印一次“B”后，十进制数应减去这一列对应的权，即$132-2^7=132-128=4$。

再继续判断，十进制的 4 与第 7 列对应权2^6比较，因$4<2^6$，所以不打印，空一格。

如此继续下去：

$4<2^5$，不打印，空一格，第 6 列空格

$4<2^4$，不打印，空一格，第 5 列空格

$4<2^3$，不打印，空一格，第 4 列空格

此时，打印针头指向第 3 列，十进制数 4 等于第 3 列的权$2^2=4$，打印一个“B”字符，并减去4，最后十进制数变为0,标志这一行打印结束，而转向下一行继续处理。

这样，打印出来的情况，和汉字“计”的第一行信息一样。

对于每一个汉字的其余行以及所有汉字都按上述方法循环处理。从而实现了汉字的再现和输出打印。

为便利程序的阅读，我们除加注中文说明外，还有两点提及：一是安排了虚读技术，若不符合字库中汉字特征字时，通过虚读跳过汉字编码，继续查找；二是用了RESTORE语句，使数据区指针恢复到初始位置，从而保证下一次汉字输入时识别过程从头开始。

程序清单（见H68）和几个实例：

```
]LIST,200

2  REM H68
5  REM  汉字原理程序
10 L = 12:C = 9
20  DIM D(L)
40  PRINT "输入字的个数:"
50  INPUT W
60  FOR I = 1 TO W
65  PRINT : PRINT
70  PRINT "输入字的代码:"
75  PRINT : PRINT
80  INPUT S$
85  PRINT : PRINT : PRINT
90  READ H$
100  IF H$ = S$ THEN 180: REM 找到特征字
105  REM 110-160虚读
110  FOR P = 1 TO 12
130  READ F: NEXT P
160  GOTO 90
170  REM 180-290八一十进制转换
180 E =  INT (C / 3) - 1
190  FOR J = 1 TO L
200  READ D(J)
210 M = D(J) / 10 ^ E
220 S = 0
230  FOR K = E TO 0 STEP  - 1
240 N =  INT (M)
250 S = S + N * 8 ^ K
260 M = (M - N) * 10
270  NEXT K
```

```
280 D(J) = S
290  NEXT J
300  RESTORE
305  REM 310—410二—十进制转换
310  FOR J = 1 TO L
320  FOR K = C - 1 TO 0 STEP  - 1
330  IF D(J) >  = 2 ^ K THEN 360
340  PRINT  TAB( 10);" ";
350  GOTO 390
360  PRINT  TAB( 10);"B";
370 D(J) = D(J) - 2 ^ K
380  IF D(J) = 0 THEN 400
390  NEXT K
400  PRINT
410  NEXT J
420  NEXT I
430  END
435  REM 440-800放特征字和八进制数
440  DATA JI1,205,105,5,705,138,105,105,105,105,
105,145,105
460  DATA  SUAN,205,357,778,105,175,105,175,105,
175,51,778,051
500  DATA DIAN,21,21,778,422,422,778,422,422,778
,21,21,37
600  DATA JI2,117,113,113,753,113,353,513,113,11
3,113,113,114
710  DATA  ET,21,21,778,422,422,778,422,422,778,
21,21,37
```

```
720  DATA  ZE,377,005,11,21,21,21,778,21,21,21,1
21,61
730  DATA  NG,21,778,21,778,506,452,576,422,576,
422,422,422
740  DATA  NIE,51,111,338,546,106,127,127,127,14
6,105,115,105
750  DATA  XUE,442,223,105,778,402,576,11,778,21
,21,61,21
760  DATA  AI,377,243,125,778,422,377,101,375,45
1,21,51,108
770  DATA  KE,53,707,103,753,107,343,538,103,103
,103,103,103
780  DATA  WO,51,115,313,511,778,111,153,115,311
,525,304,102
790  DATA  JN,21,778,001,778,402,402,778,223,223
,422,522,422
800  DATA  G,21,21,778,21,21,51,51,105,105,203,2
02,402
```

2. 打印区位码

CEC－I型中华学习机有两种汉字输入法：拼音输入法和区位码输入法。

在汉字系统状态下，按下F3键，屏幕底部状态提示行出现“区位”两字，此时表明已进入区位码输入方式。

区位码输入方式是使用国家标准GB 2381-80区位方式输入汉字。方法是用0—9共10个数字键作为汉字的键入码，通过敲入4位数字键将国家标准中所规定的汉字和圆形符号显

```
]RUN
输入字的个数:
?5
```

```
输入字的代码:

?ET

    B
    B
BBBBBBBBB
B   B   B
B   B   B
BBBBBBBBB
B   B   B
B   B   B
BBBBBBBBB
    B
    B
    BBBB
```

```
输入字的代码:

?ZE

 BBBBBBB
      B
     B
    B
    B
    B
BBBBBBBBB
    B
    B
    B
  B B
   BB
```

```
输入字的代码:

?JI1

 B    B
  B   B
      B
BBB   B
  B BBBBB
  B   B
  B   B
  B   B
  B   B
  B   B
  BB  B
  B   B
```

```
输入字的代码:

?SUAN

 B    B
 BBB BBB
BBBBBBBBB
  B   B
  BBBBB
  B   B
  BBBBB
  B   B
  BBBBB
   B B
BBBBBBBBB
   B B
```

```
输入字的代码:

?JI2

  B  BBB
  B  B B
  B  B B
BBBB B B
  B  B B
 BBB B B
B B  B B
  B  B B
  B  B B
  B  B B
  B  B B
  B  B BB
```

示到光标位置上。例如，键入1706，1720后，显示“宝贝”两字。

这种输入方法对于不太熟悉拼音的使用者有不少方便，只要查到相应汉字的区位码，并在“区位”状态下敲入代码，即可得到有关汉字。同时，相对而言输入速度较快，还有一个突出的优点是，对一些特殊的汉字，拼音状态下可能查找不到，而用区位码输入则可解决问题。

例如，汉字“女”这个字，其拼音是NÜ，在用拼音输入法时，键入NÜ却查不到。而它的区位码是3714，在区位码输入时即能顺利查到。又如汉字“律”这个字，它的拼音是LÜ，同样在拼音状态下查不到，而在区位状态下敲入3441即可顺利查找。

因此，区位码输入汉字仍受到不少使用者的欢迎。为方便用户，如能设计出一个区位码和汉字相对应的区位码表，类似一个字典一样，实为需要。

如何用程序编制一个区位码的字典呢？表面上看比较难，好象它几乎无从下手，实际上它还是有规律可寻的。

一个最简单的事实是，用字符串函数：

```
PRINT CHR$(127)+CHR$(I)+CHR$(J)
```

只要确定 I（区号）和 J（位号），则可以打印出区位码为I J 的汉字来。其中127是定值，是中华学习机汉字中3个字节中的标识码。

例如
```
PRINT CHR$(127)+
      CHR$(46)+CHR$(30)
```
打印汉字“阿”。

可以设想，用二重循环，I 作为区号的控制变量，J 作

为位号的控制变量，则可打印出全部汉字（包括国标一、二级汉字）。循环体语句为：

PRINT CHR $(127)+CHR $(46+I)+CHR $(29+J)

由于国家标准中规定国标一级汉字从16区开始，16对应的学习机汉字内码为46，这样当I=0时，就从16区开始，同理，每个区有94个汉字，其位号为01—94，其位号01对应于学习机汉字内码为29。所以当I=J=0时，上述字符串函数式打印出来的是区位码为1601的串文字“啊”。

显然，J的终值是96，29+J=29+96=125，相应的位码是94。

汉字二级字库的最后一个区是87，对应于学习机内码是118，因此I的终值为118-46=72，由于I也是从0开始，所以终值为71。考虑到汉字库存贮区中可能还写有其它图形符号（事实也是如此），故I的终值可适当取得大一些，例如取I的终值为96。

至此，一个比较完整的打印区位码和相应汉字的程序，基本形成。H69是一种打印格式的程序及部份运行结果。

```
]LIST

1  REM H69
3  REM 打印区位码
5  POKE 2043,20
7  POKE 1787,3
8  POKE 1915,5
10  FOR I = 0 TO 96
```

```
15  PRINT  TAB( 18);"--";100 * (16 + I) + 1;
"--"
20  FOR J = 0 TO 96
40  PRINT  CHR$ (127) +  CHR$ (46 + I) +  CH
R$ (29 + J);
80  NEXT J
90  PRINT : PRINT
100  NEXT I
110  END
```

```
]RUN
```

--1601--

啊阿埃挨哎唉哀皑癌蔼矮艾碍爱隘鞍氨安俺按
暗岸胺案肮昂盎凹敖熬翱袄傲奥懊澳芭捌扒叭
吧笆八疤巴拔跋靶把耙坝霸罢爸白柏百摆佰败
拜稗斑班搬扳般颁板版扮拌伴瓣半办绊邦帮梆
榜膀绑棒磅蚌镑傍谤苞胞包褒剥

--1701--

薄雹保堡饱宝抱报暴豹鲍爆杯碑悲卑北辈背贝
钡倍狈备惫焙被奔苯本笨崩绷甭泵蹦迸逼鼻比
鄙笔彼碧蓖蔽毕毙毖币庇痹闭敝弊必辟壁臂避
陛鞭边编贬扁便变卞辨辩辫遍标彪膘表鳖憋别
瘪彬斌濒滨宾摈兵冰柄丙秉饼炳

--1801--

病并玻菠播拨钵波博勃搏铂箔伯帛舶脖膊渤泊
驳捕卜哺补埠不布步簿部怖擦猜裁材才财睬踩
采彩菜蔡餐参蚕残惭惨灿苍舱仓沧藏操糙槽曹
草厕策侧册测层蹭插叉茬茶查碴搽察岔差诧拆
柴豺搀掺蝉馋谗缠铲产阐颤昌猖

程序中5句为设置行允许字数，本例为一行最多20个字。7句为设置字间距，本例字与字之间相距7个点。8句为设置行距，本例行与行之间相距5个点。15句为打印的一个表头，实际上是打印的每个区的第一个起始区位号。汉字每行20个，对应的区位号是从1601开始到1620结束，下一行从1621到1640，以下类推。每一区的最后一个汉字位号为94。

但是，上述程序有明显的缺点：

其一是查找不方便，每个汉字所对应的区位号，还要使用者一个一个地去计算；

其二是当 I ＞＝12后，所有汉字与对应的区位码不符，这就造成很大的差误。

例如，在I ＞＝12后，汉字的区号相差一个区。

如汉字“健”，正确的区位码是2901，而读出的却是肌。

为此，必须对上述程序H69进一步修正，使之能打印出正确形式，并且没有任何差错，以利使用。

]RUN

1601啊 1602阿 1603埃 1604挨 1605哎
1606唉 1607哀 1608皑 1609癌 1610蔼
1611矮 1612艾 1613碍 1614爱 1615隘
1616鞍 1617氨 1618安 1619俺 1620按
1621暗 1622岸 1623胺 1624案 1625肮
1626昂 1627盎 1628凹 1629敖 1630熬
1631翱 1632袄 1633傲 1634奥 1635懊
1636澳 1637芭 1638捌 1639扒 1640叭
1641吧 1642笆 1643八 1644疤 1645巴
1646拔 1647跋 1648靶 1649把 1650耙
1651坝 1652霸 1653罢 1654爸 1655白
1656柏 1657百 1658摆 1659佰 1650败
1661拜 1662稗 1663斑 1664班 1665搬
1666扳 1667般 1668颁 1669板 1670版
1671扮 1672拌 1673伴 1674瓣 1675半
1676办 1677绊 1678邦 1679帮 1680梆
1681榜 1682膀 1683绑 1684棒 1685磅
1686蚌 1687镑 1688傍 1689谤 1690苞
1691胞 1692包 1693褒 1694剥

1701薄 1702雹 1703保 1704堡 1705饱
1706宝 1707抱 1708报 1709暴 1710豹
1711鲍 1712爆 1713杯 1714碑 1715悲
1716卑 1717北 1718辈 1719背 1720贝

```
1721钡 1722倍 1723狈 1724备 1725惫
1726焙 1727被 1728奔 1729苯 1730本
1731笨 1732崩 1733绷 1734甭 1735泵
1736蹦 1737迸 1738逼 1739鼻 1740比
1741鄙 1742笔 1743彼 1744碧 1745蓖
1746蔽 1747毕 1748毙 1749毖 1750币
1751庇 1752痹 1753闭 1754敝 1755弊
1756必 1757辟 1758壁 1759臂 1760避
1761陛 1762鞭 1763边 1764编 1765贬
1766扁 1767便 1768变 1769卞 1770辨
1771辩 1772辫 1773遍 1774标 1775彪
1776膘 1777表 1778鳖 1779憋 1780别
1781瘪 1782彬 1783斌 1784濒 1785滨
1786宾 1787摈 1788兵 1789冰 1790柄
1791丙 1792秉 1793饼 1794炳
```

仿照“打印特殊字符库”程序的模式，并稍加改动，程序如H 70所示：

```
]LIST

1  REM H70
2  REM 打印区位码
5  POKE 2043,50
7  POKE 1787,3
8  POKE 1915,5
```

```
10  FOR I = 0 TO 71
12  PRINT : PRINT
20  FOR J = 0 TO 93
28 X = 7 * (K + 1) - A
30  PRINT  TAB( X);100 * (16 + I) + (J + 1)
40  PRINT  CHR$ (127) +  CHR$ (46 + I + M)
 CHR$ (29 + J + N);
50 K = K + 1
60  IF X > = 34 THEN  PRINT :K = 0:X = X -
34
80  NEXT J
90  PRINT : PRINT
95 K = 0:A = 0
100  NEXT I
110  END
```

```
]RUN

1601啊 1602阿 1603埃 1604挨 1605哎
1606   1607唉 1608哀 1609皑 1610癌
1611蔼 1612矮 1613艾 1614碍 1615爱
1616   1617隘 1618鞍 1619氨 1620安
1621俺 1622按 1623暗 1624岸 1625胺
1626案 1627肮 1628昂 1629盎 1630
1631凹 1632敖 1633熬 1634翱 1635袄
1636傲 1637奥 1638懊 1639澳 1640芭
1641捌 1642扒 1643叭 1644吧 1645笆
```

```
1646八 1647疤 1648巴 1649拔 1650跋
1651靶 1652把 1653耙 1654坝 1655霸
1656罢 1657爸 1658白 1659柏 1660百
1661摆 1662佰 1663败 1664拜 1665稗
1666斑 1667班 1668搬 1669扳 1670般
1671颁 1672板 1673版 1674扮 1675拌
1676伴 1677瓣 1678半 1679办 1680绊
1681邦 1682帮 1683梆 1684榜 1685膀
1686绑 1687棒 1688磅 1689蚌 1690镑
1691傍 1692谤 1693苞 1694胞
```

程序H 70运行结果表明，1606，1616，1630这三个区位码对应的字符打印不出来。

不仅如此，许多区位号对应的汉字不正确。考察其原因是汉字系统中在06，16，30这三个位号上，汉字存贮空了一个单元，而所有各区在06，16，30这三个位号都有类似的情况。而在I >=12以后，所有汉字区号相差1。为此，应在程序H 70基础上，加注四个判断条件，即I =12，J=5，J=14，J =27的分别处理，完整的程序清单见打印区位码1。全部汉字区位码打印结果见本书附录2.中华学习机汉字区位码。

```
0 NEW POKE 1659,5
1 REM H70
]LIST

2  REM 打印区位码1
5  POKE 2043,50
7  POKE 1787,3
```

```
8  POKE 1915,5
10  FOR I = 0 TO 71
12  IF I = 12 THEN M = 1
20  FOR J = 0 TO 93
28 X = 7 * (K + 1) - A
30  PRINT  TAB( X);100 * (16 + I) + (J +
 1);
35  IF J = 5 THEN N = 1
36  IF J = 14 THEN N = 2
38  IF J = 27 THEN N = 3
40  PRINT  CHR$ (127) +  CHR$ (46 + I +
M) +  CHR$ (29 + J + N);
50 K = K + 1
60  IF X >  = 34 THEN  PRINT :K = 0:X =
X - 34
80  NEXT J
85 N = 0
90  PRINT : PRINT
95 K = 0:A = 0
100  NEXT I
110  END
```

区位码不仅是汉字输入方法中常用的编码方案，而且它具有使广大程序员感兴趣的独特字符，但目前市场上又没有现成的区位码本，因此在这里介绍了打印区位码的方法并提供了全部码本（详见本书十、附录)。由于国家标准区位码是一致的，故对其它微机使用区位输入也有实用价值。

调用特殊字符★，并加注区号表头，同时控制打印机，使每页打印纸上只打印两个区的区位码和汉字，然后自动跳

页，便于打印整个码本自动控制，以利存档，见程序H71：

```
]LIST

1  REM H71
2  REM  打印区位码2
5  POKE 2043,50
7  POKE 1787,3
8  POKE 1915,5
9  FOR I = 0 TO 71
11  PRINT  TAB( 19); CHR$ (127) +  CHR$
(125) +  CHR$ (124);"  ";16 + I;"  "; CH
R$ (127) +  CHR$ (125) +  CHR$ (124)
12  PRINT : PRINT
15  IF I = 12 THEN M = 1
20  FOR J = 0 TO 93
28 X = 7 * (K + 1) - A
30  PRINT  TAB( X);100 * (16 + I) + (J +
 1);
35  IF J = 5 THEN N = 1
36  IF J = 14 THEN N = 2
38  IF J = 27 THEN N = 3
40  PRINT  CHR$ (127) +  CHR$ (46 + I +
M) +  CHR$ (29 + J + N);
50 K = K + 1
60  IF X >  = 34 THEN  PRINT :K = 0:X =
X - 34
80  NEXT J
85 N = 0
90  PRINT : PRINT
```

```
95 K = 0:A = 0
98  IF I < > INT (I / 2) * 2 THEN  FOR
 S = 1 TO 15: PRINT : NEXT S
100  NEXT I
110  END
```

相应的运行结果（仅打印 1 个区）

```
]RUN
                    ★ 18 ★

1801病 1802并 1803玻 1804菠 1805播
1806拨 1807钵 1808波 1809博 1810勃
1811搏 1812铂 1813箔 1814伯 1815帛
1816舶 1817脖 1818膊 1819渤 1820泊
1821驳 1822捕 1823卜 1824哺 1825补
1826埠 1827不 1828布 1829步 1830簿
1831部 1832怖 1833擦 1834猜 1835裁
1836材 1837才 1838财 1839睬 1840踩
1841采 1842彩 1843菜 1844蔡 1845餐
1846参 1847蚕 1848残 1849惭 1850惨
1851灿 1852苍 1853舱 1854仓 1855沧
1856藏 1857操 1858糙 1859槽 1860曹
1861草 1862厕 1863策 1864侧 1865册
1866测 1867层 1868蹭 1869插 1870叉
1871茬 1872茶 1873查 1874碴 1875搽
```

1876察 1877岔 1878差 1879诧 1880拆
1881柴 1882豺 1883搀 1884掺 1885蝉
1886馋 1887谗 1888缠 1889铲 1890产
1891阐 1892颤 1893昌 1894猖

3．汉语拼音字典

用计算机编制一个汉语拼音字典，是一个十分有趣的事情。把汉字存放在以拼音字母为特征字的DATA语句中，输入汉语拼音字母，就能找到和该拼音字母相对应的所有汉字，不仅查找速度比查真正的字典快，而且一个字也不会遗漏。因此，为汉字软件的编制提供了一个有实用价值的工具。

这对不熟悉汉语拼音的用户，或者对一些根本没有学习过拼音的同志，无疑是一个福音。他们可以利用本“字典”方便迅速地查找到需要的汉字。

不仅如此，对老师，对学生，对一切和汉语打交道的人，这本字典都是有用的。

这实在太方便了，不认识的字，记不清的字，纠正错别字，只要打开机器，放上存有本程序的磁盘，就可以顺心地查找所需要的汉字，而且一查就是一大片，许多同音字也相应地被查出来。

下面先给出程序（见汉语拼音字典），再给出使用方法，最后将程序略加说明。

程序非常短，除存有汉字的数据区外，一共只有20多行。

汉字库在本程序中仅仅放了a，b两个拼音字母的汉字，这是为了说明原理，如要成为一个完整的字典，请再继续输入其余汉字。

```
]LIST

2  REM  B-OK-1
5  REM   汉语拼音字典
20  DIM A$(50)
25  ONERR  GOTO 55
30  FOR J = 1 TO 100
40  READ A$(J)
50  NEXT J
51  PRINT "-----------------------------
----------"
52  PRINT
55  INPUT B$
57  IF B$ = "#" THEN 160
58 N = 0
60  FOR J = 1 TO 21
70 N = N + 1
80  FOR I = 1 TO  LEN (A$(J)) -  LEN (B$
) + 1
90  IF B$ < >  MID$ (A$(J),I, LEN (B$))
 THEN 120
100  PRINT B$;" ";"No:";" ";N
110  PRINT "*";A$(J);"*": GOTO 135
120  NEXT I
130  NEXT J
135  RESTORE
```

```
140 PRINT "----------------------------
------------"
150 GOTO 55
160 END



200 DATA  a啊阿呵腌锕吖啊阿呵吖锕腌嗄

210 DATA ai埃挨哎唉哀皑癌蔼矮艾碍爱隘呆
捱锿霭嗳瑷砹嗳嗌嫒
220 DATA an鞍氨安俺按暗岸胺案厂广桉鹌庵
谙埯揞铵黯犴
230 DATA ang肮昂盎
240 DATA ao凹敖熬翱袄傲奥懊澳嚣鳌獒螯鏊
遨嗷廒鏖拗媪鏊骜坳拗岙
250 DATA ba芭捌扒叭吧笆八疤巴拔跋靶把耙
坝霸罢爸岜粑茇菝魃钯鲅灞
260 DATA bai白柏百摆佰败拜稗掰捭呗伯
270 DATA ban斑班搬扳般颁板版扮拌伴瓣半
办绊瘢癍坂钣舨阪
280 DATA bang邦帮梆榜膀绑棒磅蚌镑傍谤浜
蒡
290 DATA bao苞胞包褒剥薄雹保堡饱宝抱报
暴豹鲍爆炮曝刨瀑龅煲孢葆鸨褓趵勹
300 DATA bei杯碑悲卑北辈背贝钡倍狈备惫
焙被埤萆鹎陂鞴蓓孛碚邶悖褙鐾呗臂
310 DATA ben奔苯本笨夯贲锛畚坌
320 DATA beng崩绷甭泵蹦迸蚌嘣甏
```

```
330  DATA  bi逼鼻比鄙笔彼碧蓖蔽毕毙毖币
庇痹闭敝弊必辟壁臂避陛拂秘泌荸吡秕俾舭匕
妣贲埤芘荜萆薜哔畀跸髀铋筚箅篦狴庳愎滗湢
裨璧襞嬖弼婢
340  DATA bian鞭边编贬扁便变卞辨辩辫遍砭
蝙笾鳊[illegible]煸碥匾窆褊苄忭汴弁缏
350  DATA biao标彪膘表勺彭杓飙镖镳飑瘭骠
裱婊鳔
360  DATA bie鳖憋别瘪蹩
370  DATA bin彬斌濒滨宾摈玢槟豳镔傧缤鬓
殡髌膑
380  DATA bing兵冰柄丙秉饼炳病并屏槟邴禀
摒疒
390  DATA bo玻菠播拨钵波博勃搏铂箔伯帛舶
脖膊渤泊驳卜剥薄柏百趵饽孛鹁礴踣钹亳跛簸
擗檗擘啵魄般
400  DATA bu捕卜哺补埠不布步簿部怖堡埔逋
晡钸醭卟钚瓿


]RUN
?bu
bu No: 21
*bu捕卜哺补埠不布步簿部怖堡埔逋晡钸醭卟
钚瓿*
-----------------------------------
?ao
ao No: 5
*ao凹敖熬翱袄傲奥懊澳嚣聱獒螯鳌遨嗷廒鏖
拗媪鏊骜坳拗岙  *
-----------------------------------
```

```
?bing
bing No: 19
*bing兵冰柄丙秉饼炳病并屏槟邴禀摒疒*
------------------------------------
?a
a No: 1
*a啊阿呵腌锕吖啊阿呵吖锕腌嗄    *
--------------------------------------
?ang
ang No: 4
*ang肮昂盎*
--------------------------------------
?sa
--------------------------------------
?BING
--------------------------------------
?bo
bo No: 20
*bo玻菠播拨钵波博勃搏铂箔伯帛舶脖膊渤泊
驳卜剥薄柏百趵悖孛鹁礴踣钹亳跛簸掰檗擘啵
晚般*
--------------------------------------
?#
```

本字典的操作方法简便灵活，甚至根本不熟悉计算机语言的人，几分钟以内就可以学会操作，无师可以自通。

敲入RUN并回车后，屏幕上出现一个?,此时只要键入小写的拼音字母，实际上是英语字母，即可查找到对应该拼音字母的所有同音汉字。甚至敲入一个字母，也可以查找到

对应这个字母的汉字。

由于本程序是一个原理程序，DATA语句中只放了a和b打头的汉字，所以不是a，b打头的字母查找不到。实用时请将所有汉字连同拼音字母存入DATA区中。如sa查不到。

特征字用小写英文字母，输入大写字母查不到。如小写bing能查到，BING查不到。若要继续查找，键入#并回车即可。

开机后可以查找任意次，而不需要经任何转换。

如果删去 110 句后面的GOTO135，则又是一种检索方法，请见运行实例：

```
]RUN
?ao
ao No: 5
*ao凹敖熬翱袄傲奥懊澳嚣聱獒螯鳌遨嗷廒鏖
拗媪鏊骜坳拗岙   *
ao No: 10
*bao苞龅包褒剥薄雹保堡饱宝抱报暴豹鲍爆刨
曝刨瀑龅煲孢葆鸨褓趵宀门*
ao No: 16
*biao标彪膘表勺彭杓飙镖镳飑瘭骠裱婊鳔*
----------------------------------------
?ban
ban No: 8
*ban斑班搬扳般颁板版扮拌伴瓣半办绊瘢癍坂
钣舨阪*
ban No: 9
```

```
*bang邦帮梆榜膀绑棒磅蚌镑傍谤浜蒡*
------------------------------------------

?bi
bi No: 14
*bi逼鼻比鄙笔彼碧蓖蔽毕毙毖币庇痹闭敝弊
必辟壁臂避陛捭秘泌荸吡秕俾舭匕妣贲埤芘荜
萆薜哔畀跸髀铋筚箅篦狴庳愎滗濞裨璧襞嬖弼
婢  *
bi No: 15
*bian鞭边编贬扁便变卞辨辩辫遍砭蝙笾鳊贬
煸碥匾窆褊苄忭汴弁缏*
bi No: 15
*bian鞭边编贬扁便变卞辨辩辫遍砭蝙笾鳊贬
煸碥匾窆褊苄忭汴弁缏*
bi No: 16
*biao标彪膘表勺髟杓飙镖镳飑瘭骠裱婊鳔*
bi No: 17
*bie鳖憋别瘪蹩*
bi No: 18
*bin彬斌濒滨宾摈玢槟豳镔傧缤鬓殡髌膑*
bi No: 19
*bing兵冰柄丙秉饼炳病并屏槟邴禀摒疒*
------------------------------------------

?BI
------------------------------------------

?#
```

运行实例说明:

· 敲入ao，能查找到拼音为ao，bao，biao的所有汉字。

· 敲入ban，能查找到ban，bang的所有拼音相对应的汉字。

· 敲入bi，能查到bi，bian，biao，bie，bin，bing的拼音汉字。

· 敲入大写BI，查找不到，以虚点线提示找不到信息。

· 敲入#结束。

这就是说，敲入某个或某几个拼音字母，则可以查找到包含该字母的所有汉字。这样做，有优点也有缺点，优点是可以给出更多的汉字供选择，如ban中没有的汉字，bang中就有可能有。缺点是查找的太多了，有的还重复显示几次。如bian就显示了两次。而当汉字库中包含了全部汉语拼音字典时，反而不易辨认，需按CTRL-S键边查边看。

所以实际使用时，还是在110句后面加上GOTO135为好。

本程序编制要点说明：

① DIM语句是数组说明语句，本程序用字符串A$作为变量名，下标用了50，这样只可以读50组DATA区中的信息，实用编制一个完整的字典时，下标要用到400多个，如DIMA$(450)。如本程序DATA语句共21个，每个DATA语句存放一种拼音的汉字，若用DIMA$(10)，则从A$(11)到A$(20)的信息读不到。

② 每一个DATA语句只存放一个同音汉字的拼音代码，和该代码对应的汉字。这样做，输入汉字方便，容易检查。而排列顺序按字典格式先后顺序排放。

③ 为了能连续查找，135句的恢复数据区语句RESTORE是必不可少的，否则J循环后，经150句GOTO55上去后，

将无数据可读。

④ J循环变量的终值，应和DATA语句中存放的汉字信息一致，不能少也不要多，少了则有些信息读不到，多了则浪费时间，查找速度变慢。这里系指60—130句的J循环。

⑤ 30句的J循环变量终值，可以取得大一些，原则上大于DATA语句的汉字信息个数都不会出问题，因为有25句ONERR GOTO语句的安排，程序不会因READ/DATA不配对而出错。J的终值放的大一点，其好处是可以使DATA区信息增多留有余量。

⑥ 输入特征字(或称关键字）一定要小写，目的防止和汉字信息的内码相混淆，否则改为大写，会查找出一些莫名其妙的信息来。

⑦ DATA语句中特征字和汉字输入方法，进入中文状态后，按小写字母键送入小写拼音字母至DATA语句后，按F2键进入拼音输入方式，再键入对应的小写字母，然后按1,2,3,4,5,6键，这样，对应的汉语拼音小写字母的汉字将一个一个地跳入 DATA 语句中，按“＞”键，直到对应于输入的拼音字母的全部汉字输入完了，按RETURN键，即可存贮一个DATA语句汉字信息。以后的输入同上方法。

20—160句的原理，在本书中有几次提到，这里从略。

七、现 代 管 理

1. 最少余料

实际生活或工作中，常常碰到用什么方法截取才使用料最省，例如裁衣、裁纸、锯木料、截断钢铁等等。通常的方法都是根据手工算，或者反复用尺量，这样，既费时又费工，还不能保证最大限度的节省原材料。如果编制一个程序，利用条件语句判断，循环语句查找投试，就方便多了。下面是一个应用实例，读者不难以本程序的设计方法为参考，编制出更多地解决各种各样的实际问题程序来。

一根长369cm的钢筋，截成长为69cm和39cm两种规格的短料。问在这两种规格至少各截一根的前提下，如何截法所剩余料最少？

解法1，设X，Y分别表示69cm，39cm两种规格的根数。

X1，Y1分别表示最佳方案时两种规格的根数。

Z，D为两种规格相邻方案的余料。

程序设计思想是让Y从1递增，X逐一递减，根据方程Z＝369－69X－39Y决定余料Z，将Z值与D值（D＝369为开始值）比较，Z＞D循环再找；Z＜D进行交换，即D＝Z：X1＝X：Y1＝Y，再循环上去，从而最后寻找出最佳方料使余料最少。见程序H72：

```
]LIST

5  REM H72
10  REM  最少余料1
20 Y = 0
30 D = 369
40 Y = Y + 1
50 X =  INT ((369 - Y * 39) / 69)
60 Z = 369 - X * 69 - Y * 39
70  IF D < Z THEN 90
80 D = Z:X1 = X:Y1 = Y
90  IF X > 1 THEN 40
100  PRINT "最少余料:";D;"    X=";X1;"
  Y=";Y1
110  END


]RUN
最少余料:6    X=3    Y=4
```

解法2，据题意，容易看出在Y最少取一根的情况下，X最多只能取4根；同理，在X最少取一根时，Y最多只能取7根。这样，可以安排两重循环处理，X、Y的初值和终值分别为1，4和1，7，循环体中安排D＝369－69X－39Y，从而有一个X和Y值，就有一个D值，程序运行后，计算机会自动算出各种可能情况下的X，Y和D值。我们可以根据结果,稍加判断即可得出X＝3，Y＝4，D＝6的结果是最佳

方案。程序见最少余料 2，其中负数显然是不合理的。本程序的最大优点在于，可以提供多种截取根数的方案，从而可以为实际工作时选用，例如，除 X = 3，Y = 4，D = 6以外，X = 4，Y = 2，D = 15；X = 1，Y = 7，D = 27以及 X = 2，Y = 5，D = 36都是可以选用的，因为余数 D值不可以再作本题要求，不再可以截取。见程序 H73：

```
]LIST

2  REM H73
5  REM  最少余料2
10  PRINT  TAB( 1);"X"; TAB( 5);"Y";" 最
少余料"
20  FOR X = 1 TO 4
30  FOR Y = 1 TO 7
40 D = 369 - 69 * X - 39 * Y
50  PRINT X;"   ";Y;"   ";D
60  NEXT Y
70  NEXT X
80  END



]RUN
X   Y 最少余料
1   1   261
1   2   222
1   3   183
1   4   144
1   5   105
1   6   66
```

```
1  7  27
2  1  192
2  2  153
2  3  114
2  4  75
2  5  36
2  6  -3
2  7  -42
3  1  123
3  2  84
3  3  45
3  4  6
3  5  -33
3  6  -72
3  7  -111
4  1  54
4  2  15
4  3  -24
4  4  -63
4  5  -102
4  6  -141
4  7  -180
```

解法3，程序H73运行后，D值出现许多负值，说明上述程序不够完善，事实上只要加一个条件判断语句，当D小于零时再循环查找，程序如H74：

```
]LIST

2  REM H74
5  REM   最少余料3
```

```
10  PRINT  TAB( 1);"X"; TAB( 5);"Y";" 最
少余料"
20  FOR X = 1 TO 4
30  FOR Y = 1 TO 7
40 D = 369 - 69 * X - 39 * Y
45  IF D < 0 THEN 60
50  PRINT X;"   ";Y;"   ";D
60  NEXT Y
70  NEXT X
80  END
```

```
]RUN
X   Y 最少余料
1   1   261
1   2   222
1   3   183
1   4   144
1   5   105
1   6   66
1   7   27
2   1   192
2   2   153
2   3   114
2   4   75
2   5   36
3   1   123
3   2   84
3   3   45
3   4   6
4   1   54
4   2   15
```

解法 4，实际上，本题解法不一定用二重循环，因为当Y值取一系列值后，X就有一一对应的值，X值的循环可以用赋值语句来实现，这样，循环次数可以大大减少，提高了程序执行的速度。见程序H75：

```
]LIST

2  REM H75
5  REM 最少余料4
10  PRINT  TAB( 1);"X"; TAB( 5);"Y";" 最
少余料"
20  FOR Y = 1 TO 7
30 X = 7 - Y
40 D = 369 - 69 * X - 39 * Y
45  IF D < 0 THEN 60
50  PRINT X;"   ";Y;"   ";D
60  NEXT Y
80  END



]RUN
X   Y 最少余料
3   4   6
2   5   36
1   6   66
0   7   96
```

解法 5，以上几个程序中除解法 1 外，都有几种解，而真正符合题意要求的最佳余料应只有一组，还需要我们再判断，实在不方便，其实这种判断完全可以由计算机自动完成。

例如当余料 Z 小于 0 时，显然不合理；同时最少余料 D 小于相邻余料 Z 时，也不合要求。考虑到这两种情况后，程序可以这样编写，见H76：

```
]LIST

2  REM H76
5  REM  最少余料5
10  PRINT  TAB( 1);"X"; TAB( 5);"Y";" 最
少余料"
15 D = 369
20  FOR X = 1 TO 4
30  FOR Y = X TO 7
40 Z = 369 - 69 * X - 39 * Y
50  IF Z < 0 THEN 70
55  IF D < Z THEN 70
60 D = Z:X1 = X:Y1 = Y
70  NEXT Y
80  NEXT X
90  PRINT X1;"   ";Y1;"   ";D
95  END


]RUN
X   Y 最少余料
3   4   6
```

2. 最佳运输方案

某公司有两个下属单位 A 和B，共需某种型号的微机22

台，但这种机器只有C、D两地有存货，分别为8台和14台已知这种小型机的运费如表 7.1：

表 7.1 （单位:元/台）

终点 运费 起点	A 单 位	B 单 位
C 地	30	20
D 地	60	80

问A、B两单位如何分配机器（最少8台，最多14台），才能使总运费最少？

解：让计算机比较各种不同的分配方案，从而求出最佳的运输方案。

令X、Y为A单位从C地、D地调入的计算机，则B单位从C地、D地调进的计算机为8-X台和14-Y台，H为累计的总运费，Z为一个假想的调运方案的运费，如Z取2000(此值应取得高一点，应大于等于最高运费)。

由于每个单位究竟调进（分配）多少数量的机器是待求的，所以可以安排循环遍找，最少是8台，最多是14台，这就是循环的初值和终值。

由于C地仅有8台机器，A单位从C地可以不调(为0)，也可以调（最多是8）。因此，循环变量X的取值为0到8；同理，由于D地共有14台微机，A单位从D地调进机器是0到14中的任一个数。

至于运费，应根据从C地和D地调入的机器数及运往不

同的地点的运价来决定，即：

$$H = 30X + 60Y + 20(8 - X) + 80(14 - Y)$$

$$= 10X - 20Y + 1280$$

此值应为最低。

故有程序H77：

```
]LIST

1  REM H77
2  REM   最佳运输方案1
3  REM G-T-1
5 Z = 2000
7  FOR N = 8 TO 14
10  FOR X = 0 TO 8
20  FOR Y = 0 TO 14
30  IF X + Y < > N THEN 90
40 H = 10 * X - 20 * Y + 1280
45  IF H > Z THEN 90
50 X1 = X:Y1 = Y:Z = H
90  NEXT Y
100  NEXT X
110  PRINT "A从C地调进:";X1
115  PRINT "A从D地调进:";Y1
120  PRINT "B从C地调进:";8 - X1
125  PRINT "B从D地调进:";14 - Y1
130  PRINT "最低运费:";Z
132  PRINT
135  NEXT N
138  PRINT
140  END
```

```
]RUN
A从C地调进:0
A从D地调进:8
B从C地调进:8
B从D地调进:6
最低运费:1120

A从C地调进:0
A从D地调进:9
B从C地调进:8
B从D地调进:5
最低运费:1100

A从C地调进:0
A从D地调进:10
B从C地调进:8
B从D地调进:4
最低运费:1080

A从C地调进:0
A从D地调进:11
B从C地调进:8
B从D地调进:3
最低运费:1060

A从C地调进:0
A从D地调进:12
B从C地调进:8
B从D地调进:2
最低运费:1040

A从C地调进:0
A从D地调进:13
```

```
B从C地调进:8
B从D地调进:1
最低运费:1020

A从C地调进:0
A从D地调进:14
B从C地调进:8
B从D地调进:0
最低运费:1000
```

从以上结果可以看出，当A单位从D地调进14台微机，而B单位从C地调进8台微机时，其运费最低。故按A单位分配14台微机B单位分配8台微机处理，公司只要拿出最低运费1000元。

讨论：

若A单位分配10台微机，B单位分配12台微机；和A单位分配12台微机，B单位也分配10台微机，这两种分配方案中，那一种分配方案才能使运费最少？

显然，可以从上述不同方案中，找出合理的调运方案，前者H＝1080元，后者H＝1040元，故两单位分配相同数目微机，运费最少。

事实上，也可以用程序H78求解：

```
]LIST

1  REM H78
3  REM  G-T
```

```
5 Z = 2000
7  INPUT "N=";N: PRINT "N=";N
8  IF N =  - 1 THEN  END
10  FOR X = 0 TO 8
20  FOR Y = 0 TO 14
30  IF X + Y < > N THEN 90
40 H = 10 * X - 20 * Y + 1280
45  IF H > Z THEN 90
50 X1 = X:Y1 = Y:Z = H
90  NEXT Y
100  NEXT X
110  PRINT "A从C地调进:";X1
115  PRINT "A从D地调进:";Y1
120  PRINT "B从C地调进:";8 - X1
125  PRINT "B从D地调进:";14 - Y1
130  PRINT "最低运费:";Z
132  PRINT
135  GOTO 7

]RUN
N=10
N=10
A从C地调进:0
A从D地调进:10
B从C地调进:8
B从D地调进:4
最低运费:1080


N=12
N=12
```

```
A从C地调进:0
A从D地调进:12
B从C地调进:8
B从D地调进:2
最低运费:1040

N=-1
N=-1
```

这个结果，和程序H77的运算结果是一致的，从这里看出程序H78通用性更好一些，只要输入某一单位需要微机台数N，即可找到最佳调运方案使运费最少。

程序H78只用了二重循环，并改用INPUT语句，使程序更为灵活。第8句终止标记的安排也是一个小小的技巧。

若由于某种原因，例如与C地已事先订好合同，必须从C地买进一台微机，则程序如何安排呢？

下列两种方法之一均可：

(1) 用H77程序：

10句改为：10 FOR X=1 TO 8

或10 X=1，删去100句 NEXT X

(2) 用H78程序：

10句改为：10 FOR X=1 TO 8

或10 X=1，删去100句 NEXT X

你能说出道理吗？特别是删去FOR/NEXT循环，而改用10 X=1。请读者自行考虑，并上机验证。

如果各地不仅运费不同，而且微机价格有异，程序又如何编制呢？

设最少从C地购买一台微机，其余情况参照表 7.2 完成设计：

表 7.2

购货单位 运费 价格 供货地点	A 需购10台	B 需购12台
C可供 8 台	3200 / 30	2400 / 20
D可供14台	2800 / 60	2700 / 80

设计思路和方法同上，只需将Z值和H值改变。

下面给出两种程序H79和H80。工作原理请读者自行分析，注意程序H80中10句的安排技巧。

```
]LIST

2  REM H79
3  REM   G-9
5 Z = 100000
10  FOR X = 1 TO 8
20  FOR Y = 0 TO 14
30  IF X + Y < > 10 THEN 90
40 H = 3230 * X + 2860 * Y + 2420 * (8 -
```

```
 X) + 2780 * (14 - Y)
45  IF H > Z THEN 90
50 X1 = X:Y1 = Y:Z = H
90  NEXT Y
100  NEXT X
110  PRINT "A:C=";X1;" ";"D=";Y1
120  PRINT "B:C=";8 - X1;" ";"D=";14 - Y
1
130  PRINT "H=";Z
140  END
```

```
]RUN
A:C=1 D=9
B:C=7 D=5
H=59810
```

```
]LIST

2  REM H80
3  REM    G-10
5 Z = 100000
8 X = 1
10  FOR Y = 8 - X TO 12
30  IF X + Y < > 10 THEN 90
40 H = 3230 * X + 2860 * Y + 2420 * (8 -
 X) + 2780 * (14 - Y)
45  IF H > Z THEN 90
50 X1 = X:Y1 = Y:Z = H
90  NEXT Y
```

```
110  PRINT "A:C=";X1;" ";"D=";Y1
120  PRINT "B:C=";8 - X1;" ";"D=";14 - Y
1
130  PRINT "H=";Z
140  END
```

```
]RUN
A:C=1 D=9
B:C=7 D=5
H=59810
```

3. 最优生产顺序

计算机用于工业企业的实际生产管理已不泛其例。而且越来越发挥它的威力。如各种工程设计往往需要多方案比较，可行性研究。使用计算机不仅大大减少繁重的工作量，而且又有可能提供最佳方案，缩短设计周期，提高经济效益。下面，我们举一简单应用实例。

题目：设有n个产品经由M部机器加工，且每个产品需先后由A、B、C……机器依次加工完成；这M部机器对每个产品加工所用机时各不相同。试确定出n个产品的最优加工次序，使总加工时间为最短。

分析：这属于排序问题。根据工作的轻重缓急和经济效益等具体情况，应用数学方法来分析，利用计算机来求解，在各种可行性的生产顺序方案中，求一种最佳的生产顺序。

由于实际情况是多方面的，解决各种类型的排序问题还比较困难，为此，常根据某种特定要求来进行程序设计，如怎样缩短生产周期，提高机床利用率等。

我们先看一个最简单的实例。设有2个产品，每个产品分别由A、B两部机器加工，每部机器对各个产品的加工机时如表7.3：

表 7.3

产品号次	机器A	机器B
1	5	2
2	3	1

按上述顺序加工，则当第一种产品加工时，机器B是停机待工的；等到A机器加工完第一种产品时，机器B实际上等了5小时，而当机器A加工第二种产品时，机器B才开始工作并加工第一种产品，而等到机器B工作2小时后，第二种产品已经由机器A加工完成，必须再等1小时，待机器B完成第一类产品加工后，再送入B机器加工。这样，加上第二类产品在机器B上的工作时间，B机器共花费工时：5(停工)+4（第一类产品用时）+1（第二类产品用时）=10小时。

若上述生产顺序稍加改动，即首先加工第一种产品，然后再加工第二类产品，则B机器所花工时共有：3(停工时间)+1（第二类产品用时）+4（停工时间）+4(第一类产品工作时间）=12小时。

显然，第一种方案为优。

如果有6种产品，经由两部机器顺序加工，工时如表7.4，

你能找出完成产品加工总工时数为最短的方案吗？

表 7.4

产品号次 I	机器 A pi	机器 B gi
1	5	4
2	3	1
3	5	2
4	4	6
5	6	5
6	3	6

这比较复杂一点了，但还是可能计算出来的，仿照上面分析方法，不难求出总加工时间为34小时。

即：A机加工 1 号产品到完成，B机停工 5 小时；
A机完成 2 号产品，B机加工1号产品3小时；
A机加工 3 号产品 2 小时，B机完成 2 号产品又花去 1 小时；
A机加工 3 号产品完成，B机停工 3 小时；
A机加工 4 号产品 2 小时，B机完成 3 号产品加工任务，用时 2 小时；
A机完成 4 号产品，B机停工 2 小时；
A机完成 5 号产品，B机完成 4 号产品花 6 小时；
A机完成 6 号产品，B机加工 5 号产品 3 小时；
A机停机，B机再工作 2 小时完成 5 号产品。

最后B机再工作 6 小时，完成 6 号产品加工任务，最后也停机。

到此为止，我们仅仅只分析一种方案，并且不知道这种方案是否最佳，若改变加工次序，并找出一个最佳方案，仅用上述方法逐个分析，不仅复杂且费时甚多。

如果产品号次增多，或者每种产品经过的工序增加，则这种判断，有时几乎成为不可能。

为此，我们只有求助于计算机，协助我们做出复杂的判断了。

如何确定上述 6 种产品的最佳加工序次，使得完成产品加工的总工时为最短。

首先列出数学表达式：

$$S = \sum_{i=1}^{n} gi + \sum_{i=1}^{n} Xi$$

式中：S ＝总工作时间；gi 为第二项工作（第二道工序）在第 i 个产品上的作业时间；Xi 为第二项工作（第二道工序）在第 i 个产品上的停歇时间。

由于 $\sum_{i=1}^{n} gi$ 固定不变，故欲使总工作时间S 最短，则必须使 $\sum_{i=1}^{n} Xi$ 为最小值。

对于这类问题的排序，要用换行法求解，步骤如下：

第一步：将所有产品作业时间按一定次序排列成一个矩阵表（如上表）。表中元素按加工对象，第一项（A 机加工）第二项（B 机加工）排列；

第二步：在所有元素中选择一个最小值，如有二个以上元素的最小值相同时，则任选其一；

第三步：如最小值为pi，则将第 i 行换置在表的最上一行，如最小值是gi，则将第 i 行换置在表的最下一行；

第四步：对未换过的行的元素，再找最小值；

第五步：重复第三、四步，直至全部元素均换过行为止。

最后得到的行的次序如表7.5，这是一个最优化的序列：

表 7.5

i	pi	gi
6	3	6
4	4	6
5	6	5
1	5	4
3	5	2
2	3	1

不难分析，按上述优化作业次序进行时，其总工时为27小时，较之前者（非优化序列加工顺序）缩短工时 6 小时。

现在，我们给出计算机求解的框图，(粗框图)，见图7.1：

为实现上述设计思想，在程序上我们可以分三段来安排：

第一段：读存作业顺序的矩阵（它是非优化的加工序列）见程序10—70行，I（I）为产品号次，P（I）为A机器上作业时间，Q（I）为B 机器上作业时间(I = 1，2，……6)。

第二段：将A机器上作业时间由短到长排序（即作业时间最少的放在最上面，作业时间最多的安置在最下面）。

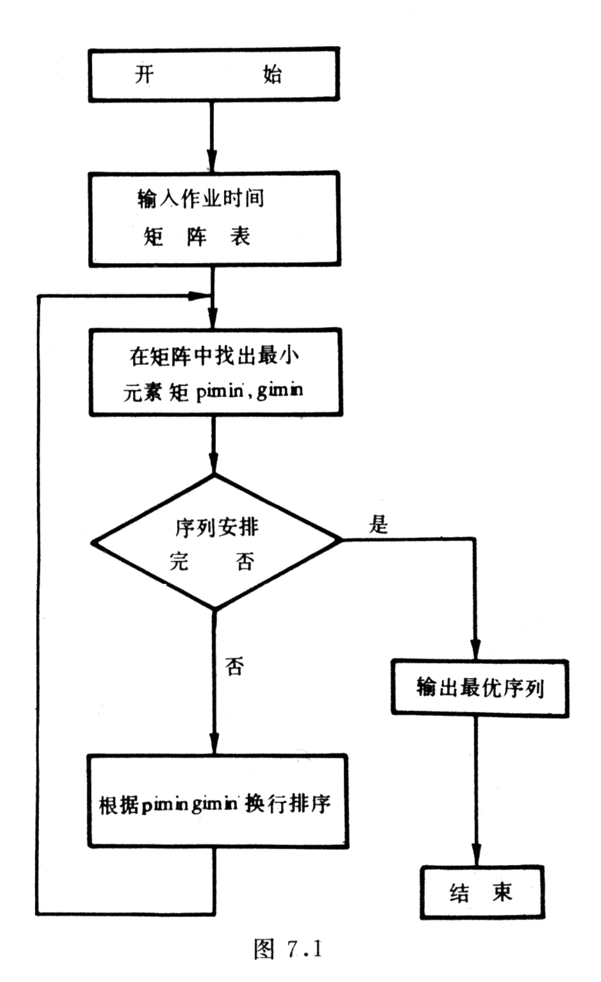

开 始
输入作业时间
矩 阵 表
在矩阵中找出最小
元素矩 pimin, gimin
序列安排
完 否
是
否
输出最优序列
根据pimin gimin 换行排序
结 束

图 7.1

注意矩阵的第1列，第3列也跟着顺序交换。见程序110—200，打印出结果是为观察执行过程。

第三段：将B机器上作业时间由长到短排序（即在B机器上费时最多的排在最上面，而费时较少的则排在最下面）。同样注意短阵的第1列，第2列同时顺序交换。见程序210—420行。最后打印出来的矩阵表系最优作业序列。

程序（见H81）及运行结果如下：

```
]RUN
----------
!I  Pi Qi!
----------        ----------        ----------
! 1  5  4!        ! 2  3  1!        ! 6  3  6!
----------        ----------        ----------
! 2  3  1!        ! 6  3  6!        ! 4  4  6!
----------        ----------        ----------
! 3  5  2!        ! 4  4  6!        ! 5  6  5!
----------        ----------        ----------
! 4  4  6!        ! 1  5  4!        ! 1  5  4!
----------        ----------        ----------
! 5  6  5!        ! 3  5  2!        ! 3  5  2!
----------        ----------        ----------
! 6  3  6!        ! 5  6  5!        ! 2  3  1!
----------        ----------        ----------
]LIST,165

2 REM H81
5 REM   最优生产顺序
8 REM :10-70句,读取作业次序矩阵
```

```
10  DIM I(6),P(6),Q(6)
12  GOSUB 400
15  PRINT "!I";" ";" Pi";" ";"Qi!"
20  FOR I = 1 TO 6
30  READ I(I),P(I),Q(I)
35  GOSUB 400
40  PRINT "! ";I;"  ";P(I);"  ";Q(I);"!"
50  NEXT I
55  GOSUB 400
60  PRINT
70  DATA 1,5,4,2,3,1,3,5,2,4,4,6,5,6,5,6
,3,6
100  REM :110-200句,将A机器作业时间排序
110  FOR I = 1 TO 5
120  FOR J = I + 1 TO 6
130  IF P(I) < = P(J) THEN 160
140 T = P(I):P(I) = P(J):P(J) = T
150 T = Q(I):Q(I) = Q(J):Q(J) = T
155 T = I(I):I(I) = I(J):I(J) = T
160  NEXT J
165  GOSUB 400
170  PRINT "! ";I(I);"  ";P(I);"  ";Q(I)
;"!"
180  NEXT I
185  GOSUB 400
190  PRINT "! ";I(6);"  ";P(6);"  ";Q(6)
;"!"
195  GOSUB 400
200  PRINT
205  REM :210-420句,将B机器作业时间排序
```

```
210  FOR I = 1 TO 5
220  FOR J = I + 1 TO 6
230  IF Q(I) >  = Q(J) THEN 260
240 T = Q(I):Q(I) = Q(J):Q(J) = T
250 T = P(I):P(I) = P(J):P(J) = T
255 T = I(I):I(I) = I(J):I(J) = T
260  NEXT J
265  GOSUB 400
270  PRINT "! ";I(I);"  ";P(I);"  ";Q(I)
;"!"
280  NEXT I
285  GOSUB 400
290  PRINT "! ";I(6);"  ";P(6);"  ";Q(6)
;"!"
300  GOSUB 400
310  END
400  PRINT "---------"
420  RETURN
```

4. 最佳配料方案

在冶金、化工、医学、农业、畜牧等各个部门，常常要求计算原料的配方，以保证产品达到一定的技术指标。同时，为了减低成本，最大限度地提高经济效益，更需要设计管理人员提供最佳的配料方案。

过去这方面的工作，常常用手算和经验估测，计算量大，花时间多，且不能保证节约原材料，经济效益也不是最好。

下面介绍一个在冶金方面的实用程序实例，浅谈一下编

制最佳配方的原理程序，其它各部门不同问题的使用，可以以此为模式，不难编制出适用不同问题的配料程序。

某钢铁厂要冶炼10吨铸铁，要求含碳量3.2—3.3%，含硅在1.9—2.0%。现有三种原料，其成份及价格如表 7.6 所示，试编制一个最佳配料方案的程序（精确到吨）。

表 7.6

原　　料	碳(C)%	硅(Si)%	单价(元/t)
生　　铁	3.8	2.8	340
废　　钢	0.6	0.2	240
废　　铜	2.6	1.4	380

分析：求取最佳配料方案，实际上是要权衡原料、成份、价格三种因素，缺一不可。三种原料究竟各取多少，才能保证铸铁的含碳(C)量和含硅（Si）量在3.2—3.3%及1.9—2.0%之间，同时，成本又是最低呢？

首先，设置三个变量X、Y、Z，让它们分别表示生铁、废钢、废铜三种原料的最佳配料的数量（精确到吨）。

据题意，X、Y、Z 应同时满足以下四条条件：

① $X+Y+Z=10$（一共炼10吨）

② $340X+240Y+380Z=$最小值（成本最低）

③ $3.2 \leqslant \frac{3.8X+0.6Y+2.4Z}{10} \leqslant 3.3$

(满足每吨铸铁含碳量的要求)

④ $1.9 \leqslant \frac{2.3X+0.2Y+1.4Z}{10} \leqslant 2.0$

(满足每吨铸铁含硅量的要求)

其次，确定三种原料的取值范围。由于题目要求精确到吨，显然，X、Y、Z的取值范围都是1到10之间，在程序上可以用循环的方法来逐步投试，循环变量就取X、Y和Z，它们的初值和终值各自为1到10。

最后，谈一下循环体的安排。

三种原料X、Y、Z必须同时满足前述四个条件时，才是本题的解。反之均不是，故可以用条件语句（IF/THEN）来判断，把它们都写在循环体中。

至于成本的最小值问题，可以这样来安排，在循环外，设置一个初值A，如A＝380×10，显然，这是一个极端情况，即取成本最高价的废铜，它应为3800元。

令最小值为B，即340X＋240Y＋380Z＝B，如果B＞A则不要（即循环再找另外的X、Y、Z值），反之，则令A＝B（此时A为最小值，即交换）。

所以程序中A、B分别表示前序方案和当前方案的配料总值（即成本数），而XI、YI、ZI为最佳配料方案时的X、Y、Z值，因此，在全部三重循环完毕，应安排打印XI、YI、ZI及A的值，则XI、YI、ZI的值即为最佳配料原料的吨数，而A值则为最低成本数，此时配方最佳，经济效益最高。

故有程序H82：

```
]LIST

5  REM  H82
10 A = 380 * 10
20  FOR X = 1 TO 10
```

```
30  FOR Y = 1 TO 10
40  FOR Z = 1 TO 10
50  IF X + Y + Z < > 10 THEN 140
60  IF (3.8 * X + 0.6 * Y + 2.4 * Z) / 10 >
3.3 THEN 140
70  IF (3.8 * X + 0.6 * Y + 2.4 * Z) / 10 <
3.2 THEN 140
80  IF (2.3 * X + 0.2 * Y + 1.4 * Z) / 10 >
2 THEN 140
90  IF (2.3 * X + 0.2 * Y + 1.4 * Z) / 10 <
1.9 THEN 140
100 B = 340 * X + 240 * Y + 380 * Z
110  IF (B > A) THEN 140
120 A = B
130 X1 = X:Y1 = Y:Z1 = Z
140  NEXT Z
150  NEXT Y
160  NEXT X
170  PRINT "X=";X1;" ";"Y=";Y1;" ";"Z=";Z1
180  PRINT "A=";A
190  END

]RUN
X=7 Y=1 Z=2
A=3380
```

程序的正确性，我们可以用手算来验证：

生铁X = 7 吨，总价为： 7 ∗ 340 = 2380元。

废钢Y = 1吨，总价为：1 * 240 = 240元。

废铜Z = 2吨，总价为：2 * 380 = 760元。

故总成本为：2380 + 240 + 760 = 3380元。

原料按上述取值后，含碳量、含硅量是否达到指标要求呢？

含碳量：(7 * 3.8 + 1 * 0.6 + 2 * 2.4)/10

= (26.6 + 0.6 + 4.8)/10

= 32.0/10

= 3.2　　　（符合要求）

同理，含硅量：(7 * 2.3 + 1 * 0.2 + 2 * 1.4)/10 = 1.91

（符合要求）

总之，利用上述程序，我们可以得到最佳配料方案。

现在，我们来研究一下，配料方案是最佳的，程序是否也是最好的呢？显然不是。

第一，程序H82是计算铸铁10吨产量的，若要计算不同产量时，必须修改程序。为不失一般性，可改用程序H83。这样做有二个好处：可以求不同产量（仍以精确到吨计算）；程序中用变量N来代替常量10(或不同产量时的其它常数值)，既可节省内存,又可加快程序的执行速度(程序H83中共有九次要用常量N = 10)。

第二，修改后的程序H83，仍有一个明显的缺点，运行速度慢，考察其原因有二点：

循环中判断语句多，加上许多小数比较，速度比较慢；

循环终值愈大，循环次数愈多，速度也愈慢。

N = 10吨时，PC-1500机运行时间为50秒；N = 20吨时，执行时间为6分10秒。

为此，我们设法减少一重循环，因为：

$X+Y+Z=N$

$\therefore Z=N-X-Y$

考虑到负值无意义，故取Z＝ABS（N－X－Y）。

原程序中的40句，FOR Z＝1TO 10用Z＝ABS（N－X－Y）代替，删去140句NEXT Z，同时各条件判断语句THEN后面的行号均改为150，则运行速度大大提高。N＝10吨时，程序运行时间为16秒；N＝20吨时，速度为57秒。程序及运行结果附后（见H84）。

第三，为了进一步提高程序运行速度，二个循环的终值还可以进一步改动。

例如：N＝10时，20、30句的循环终值N换成8，因为X＋Y＋Z＝10，每种原料至少取1吨时（如X＝1，Y＝1），则另一种原料最多取8吨（如Z＝8），这样，程序运行时间又减少4秒。

同理，N＝20时，20、30句的循环终值N换成18，则N＝20时，运行时间为52秒。

第四，为了进一步减少内存，同时又有比较高的执行速度，请用程序H85。

```
]LIST

5  REM  H83
8  INPUT "N=";N: PRINT "N=";N
10 A = 380 * N
20  FOR X = 1 TO N
```

```
30  FOR Y = 1 TO N
40  FOR Z = 1 TO N
50  IF X + Y + Z < > N THEN 140
60  IF (3.8 * X + 0.6 * Y + 2.4 * Z) / N > 3
.3 THEN 140
70  IF (3.8 * X + 0.6 * Y + 2.4 * Z) / N < 3
.2 THEN 140
80  IF (2.3 * X + 0.2 * Y + 1.4 * Z) / N > 2
 THEN 140
90  IF (2.3 * X + 0.2 * Y + 1.4 * Z) / N < 1
.9 THEN 140
100 B = 340 * X + 240 * Y + 380 * Z
110  IF (B > A) THEN 140
120 A = B
130 X1 = X:Y1 = Y:Z1 = Z
140  NEXT Z
150  NEXT Y
160  NEXT X
170  PRINT "X=";X1;" ";"Y=";Y1;" ";"Z=";Z1
180  PRINT "A=";A
190  END
```

```
]RUN
N=10
N=10
X=7 Y=1 Z=2
A=3380
```

```
]RUN
```

```
N=20
N=20
X=16 Y=3 Z=1
A=6540
```

```
]LIST

5  REM H84
8  INPUT "N=";N: PRINT "N=";N
10 A = 380 * N
20  FOR X = 1 TO N
30  FOR Y = 1 TO N
40 Z =  ABS (N - X - Y)
50  IF X + Y + Z <  > N THEN 150
60  IF (3.8 * X + 0.6 * Y + 2.4 * Z) / N > 3
.3 THEN 150
70  IF (3.8 * X + 0.6 * Y + 2.4 * Z) / N < 3
.2 THEN 150
80  IF (2.3 * X + 0.2 * Y + 1.4 * Z) / N > 2
 THEN 150
90  IF (2.3 * X + 0.2 * Y + 1.4 * Z) / N < 1
.9 THEN 150
100 B = 340 * X + 240 * Y + 380 * Z
110  IF (B > A) THEN 150
120 A = B
130 X1 = X:Y1 = Y:Z1 = Z
150  NEXT Y
160  NEXT X
170  PRINT "X=";X1;" ";"Y=";Y1;" ";"Z=";Z1
180  PRINT "A=";A
190  END
```

```
]RUN
N=10
N=10
X=7 Y=1 Z=2
A=3380
```

```
]RUN
N=20
N=20
X=16 Y=3 Z=1
A=6540
```

```
]LIST

5  REM H85
8  INPUT "N,M=";N,M: PRINT "N,M=";N;" ";M
10 A = 380 * N
20  FOR X = 1 TO N
30  FOR Y = 1 TO N
40 Z =  ABS (N - X - Y)
50  IF X + Y + Z <  > N THEN 150
60 K = (3.8 * X + 0.6 * Y + 2.4 * Z) / N
70 J = (2.3 * X + 0.2 * Y + 1.4 * Z) / N
80  IF K > 3.3 THEN 150
85  IF K < 3.2 THEN 150
90  IF J > 2 THEN 150
95  IF J < 1.9 THEN 150
100 B = 340 * X + 240 * Y + 380 * Z
110  IF (B > A) THEN 150
120 A = B
```

```
130 X1 = X:Y1 = Y:Z1 = Z
150  NEXT Y
160  NEXT X
170  PRINT "X=";X1;" ";"Y=";Y1;" ";"Z=";Z1
180  PRINT "A=";A
190  END
```

```
]RUN
N,M=10,8
N,M=10 8
X=7 Y=1 Z=2
A=3380
```

```
]RUN
N,M=20,18
N,M=20 18
X=16 Y=3 Z=1
A=6540
```

5. 网络图时间参数计算程序

网络计划技术是50年代后期发展起来的一种计划管理方法。它是以网络图的形式制定计划，求得计划的最优方案，并据以组织和控制生产，达到预定目标的一种科学管理方法。在工厂的新产品试制工作中或是各种设备的安装、改造和维修计划中以及各种基本建设计划中有较广泛的应用。

在绘制网络图的过程中，需要计算各工序的时间参数，

确定关键工作和关键路线，利用时差，改善计划，求得工期资源与成本的最优方案。

本程序为网络时间参数的计算，计算方法采用表算法原理。

程序中：

10—35　　为中文提示

40—85　　为输入各项参数；数组S（M）

存放最早开工时间参数；E（M）为

存放最早完工时间参数；L（M）为

存放最迟开工时间参数；F（M）为

存放最迟完工时间参数。

参数输入中如有错误，第80句提供修改机会。

90—110　　调用195—210子程序计算最早开工时间参数和最早完工时间参数。

115—135　　调用215—230子程序计算最迟开工时间参数和最迟完工时间参数。

140—160　　打印表头。

165—190　　打印时间参数。

195　　关打印机，结束。

运行时，按中文提示输入数据（对虚作业其工序作业时间为0）。如输入有错误可键入"0"，计算机问："需要修改哪一工序？"，键入要修改的工序号进行修改。修改完毕，再键入"0"，然后键入需要继续输入的工序号，继续往下进行。全部数据输入完毕后即可自动打印出运算结果。

程序清单（见H86）和运行实例。

运行实例：

筹建某产品的生产线工程，其作业明细表如表7.7。

表 7.7

作业序号	作业代号	作 业 名 称	先行作业	时间(d)
1	A	市场调查	无	12
2	B	产品研制	无	26
3	C	资金筹备	A	24
4	D	工程项目分析	A.B	6
5	E	产品设计	A.B	12
6	F	厂房基建	A.B	40
7	G	成本计划	C.D	8
8	H	人员筹备	C.D	18
9	I	生产计划	E.G	4
10	J	设备计划	H	10
11	K	器材筹备	I.J.F	24
12	L	人员安排	K	8
13	M	学习工艺	H	5
14	N	设备布置	F	16
15	O	调试生产	N.L.M	22

```
5  REM H86
10  PRINT "本程序可计算网络图时间参数,程序中
"
15  PRINT " M—网络图工序数  I—工序箭尾号"
20  PRINT " T—工序作业时间  J—箭头结点号"
25  PRINT "ES—最早开工时间 EF—最早完工时间
"
30  PRINT "LS—最迟开工时间 LF—最迟完工时间
"
35  PRINT " R—总时差        GJ—关键路线"
40  INPUT "请输入工序数:";M
45  DIM A(M,3),S(M),E(M),L(M),F(M),R(M)
50  FOR P = 1 TO M
55  PRINT "工序代号:";P
60  INPUT "请输入工序箭尾号:";A(P,1)
```

```
65  IF A(P,1) = 0 THEN  INPUT "需要修改哪一
工序?";P: GOTO 55
70  INPUT "请输入箭头结点号:";A(P,2)
75  INPUT "请输入工序作业时间:";A(P,3)
80  IF P = M THEN  PRINT "是否需要修改(Y/N)?
";: GET Q$: PRINT Q$: IF Q$ = "Y" THEN 55
85  NEXT P
```

```
90  FOR I = 1 TO M
95  IF A(I,1) = 1 THEN S(I) = 0:E(I) = A(I,3
): GOTO 110
100  IF A(I,1) <  > 1 THEN  GOSUB 195
105 E(I) = S(I) + A(I,3)
110  NEXT I
115  FOR I = M TO 1 STEP  - 1
120  IF I = M THEN F(I) = E(I):L(I) = S(I)
125  IF I <  > M THEN F(I) = F(M): GOSUB 215
130 L(I) = F(I) - A(I,3):R(I) = L(I) - S(I)
135  NEXT I
140  POKE 1659,5
145  PRINT  TAB( 23)"网络图时间表"
150  PRINT "———————————————————
——————————————"
155  PRINT "代号  编  号    T    ES    EF
 LS    LF     R    GJ"
157  POKE 1659,1
160  PRINT "———————————————————
——————————————"

165  FOR I = 1 TO M
170  PRINT  TAB( 1);I; TAB( 7);A(I,1);"—";A
(I,2); TAB( 16);A(I,3); TAB( 22);S(I); TAB(
```

```
28);E(I); TAB( 34);L(I); TAB( 6);F(I); TAB(
12);R(I);
175  IF R(I) = 0 THEN  PRINT  TAB( 18);A(I,1
);"—";A(I,2);
180  PRINT
185  NEXT I
190  PRINT "———————————————————————
——————————————"
192  POKE 1659,0: END
195  FOR J = 1 TO I - 1
200  IF A(J,2) = A(I,1) AND E(J) > S(I) THEN
 S(I) = E(J)
205  NEXT J
210  RETURN
215  FOR J = M TO I - 1 STEP  - 1
220  IF A(J,1) = A(I,2) AND L(J) < F(I) THEN
 F(I) = L(J)
225  NEXT J
230  RETURN

]RUN
本程序可计算网络图时间参数,程序中
 M—网络图工序数  I—工序箭尾号
 T—工序作业时间  J—箭头结点号
ES—最早开工时间 EF—最早完工时间
LS—最迟开工时间 LF—最迟完工时间
 R—总时差        GJ—关键路线
请输入工序数:16
工序代号:1
请输入工序箭尾号:1
请输入箭头结点号:3
```

```
请输入工序作业时间:12
工序代号:2
请输入工序箭尾号:1
请输入箭头结点号:5
请输入工序作业时间:26
工序代号:3
请输入工序箭尾号:3
请输入箭头结点号:7
请输入工序作业时间:24
工序代号:4
请输入工序箭尾号:5
请输入箭头结点号:7
请输入工序作业时间:6
工序代号:5
请输入工序箭尾号:5
请输入箭头结点号:9
请输入工序作业时间:12
工序代号:6
请输入工序箭尾号:5
请输入箭头结点号:11
请输入工序作业时间:40
工序代号:7
请输入工序箭尾号:7
请输入箭头结点号:9
请输入工序作业时间:8
工序代号:8
请输入工序箭尾号:7
请输入箭头结点号:13
请输入工序作业时间:18
工序代号:9
请输入工序箭尾号:9
```

```
请输入箭头结点号:15
请输入工序作业时间:4
工序代号:10
请输入工序箭尾号:13
请输入箭头结点号:15
请输入工序作业时间:10
工序代号:11
请输入工序箭尾号:11
请输入箭头结点号:15
请输入工序作业时间:0
工序代号:12
请输入工序箭尾号:15
请输入箭头结点号:17
请输入工序作业时间:24
工序代号:13
请输入工序箭尾号:17
请输入箭头结点号:19
请输入工序作业时间:8
工序代号:14
请输入工序箭尾号:13
请输入箭头结点号:19
请输入工序作业时间:5
工序代号:15
请输入工序箭尾号:11
请输入箭头结点号:19
请输入工序作业时间:16
工序代号:16
请输入工序箭尾号:19
请输入箭头结点号:22
请输入工序作业时间:22
是否需要修改(Y/N)?
```

表 7.8 网络图时间表

代号	编号	T	ES	EF	LS	LF	R	GJ
1	1—3	12	0	12	2	14	2	
2	1—5	26	0	26	0	26	0	1—5
3	3—7	24	12	36	14	38	2	
4	5—7	6	26	32	32	38	6	
5	5—9	12	26	38	50	62	24	
6	5—11	40	26	66	26	66	0	5—11
7	7—9	8	36	44	54	62	18	
8	7—13	18	36	54	38	56	2	
9	9—15	4	44	48	62	66	18	
10	13—15	10	54	64	56	66	2	
11	11—15	0	66	66	66	66	0	11—15
12	15—17	24	66	90	66	90	0	15—17
13	17—19	8	90	98	90	98	0	17—19
14	13—19	5	54	59	93	98	39	
15	11—19	16	66	82	82	98	16	
16	19—22	22	98	120	98	120	0	19—22

八、情 报 检 索

下面即将写出的程序清单，是一个具有多种用途和多种功能的检索程序。

所谓多种用途是指程序中除数据区外,主要程序段(20—160)是一个多用途的子程序。它可以查找图书目录、文献资料、库存物资、工资报表、人事档案、公安口卡、病历登记、统计报表等等。凡是涉及到社会生产、生活的各个方面，诸如生产管理科研、教学、工程技术、等各个领域，凡是需要情报资料检索的地方，都可以以本程序为模式，适当改动部份语句和数据区，即可付诸实用。

所谓多种功能，是指程序查找方法多种多样，查找手段灵活方便，使用者只要稍事熟悉，即能运用自如。

本程序的最大特点，是程序简短、易懂、通用、灵活。

现在，我们以人事档案的管理和应用为例，说明本程序的编制方法。

现列出主要程序段，见H87：

```
]LIST

10  REM H87
20  DIM A$(10)
30  FOR J = 1 TO 10
40  READ A$(J)
45  PRINT A$(J)
50  NEXT J
52  PRINT "--------------------------------
```

```
-------"
53  PRINT
55  INPUT B$
57  IF B$ = "#" THEN 160
58 N = 0
60  FOR J = 1 TO 10
70 N = N + 1
80  FOR I = 1 TO  LEN (A$(J)) -  LEN (B$) +
1
90  IF B$ <  >  MID$ (A$(J),I, LEN (B$)) THE
N 120
100  PRINT B$;" ";"No:";" ";N
110  PRINT "*";A$(J);"*"
120  NEXT I
130  NEXT J
140  PRINT "--------------------------------
--------"
150  GOTO 55
160  END
```

对于程序H87的编制要点，有以下几点说明：

(1) A＄(10)：字符串数组，存贮职工档案信息，本程序以10人为例，所以用A＄(10)，若职工人数为200人，则用A＄(200)，若超过255人，可用A＄，(255)、B＄(100)等。

(2) 30—50行，是一个循环，循环变量为J，循环次数为10，若人数为200人，则终值只要改成200即可。本程序段的功能是读存职工档案信息，并打印全体职工的信息。

(3) INPUT B＄，是输入查找的关键字。B＄的输入十分灵活，例如输入职工序号B5；输入职工的姓或名或姓名，全体或姓名中部份字符；输入性别W或M；输入年龄、文化

程度、政治面貌、健康状况等等。

(4) 57行是查找结束标志。58和70句，指明查找某人信息在存贮数据区中的位置。

(5) 60—130句中外循环（用循环变量J），目的是扫描整个数据区。80—120句是内循环，其作用是取出每个职工档案的全部信息，与关键字B $ 比较，有则打印结果，否则再找。全部循环完毕（包括I、J两重循环），找不到关键字的信息，则以————为标志。

(6) 150句GOTO55，是为了再找另一个关键字B $，若停止查询，则敲入字符“#”，停机。

(7) 请注意80句循环终值的写法和技巧，主要目的是提高比较速度，缩短查找时间。

关于职工档案信息编制方法如下（程序见H88）：

(1) 每个职工档案共有10项（实用单位可根据具体情况，自定项数），每项之间均以一个空格分隔，它们都存贮在DATA区中，其存贮格式：

280 DATA“B9 ABA M L-40G-12H-2
Z-23KO6 P-DT-2”

这里是采用的字符串形式存贮，各项之间不用逗号分隔，仅以空格分隔（当然也可以不用空格而作为一个连续的字符串形式存贮）。

(2) 检索代码：

序号——B，如B 9 代表查找编号为 9 的人。

姓名——按实际人员姓名填写，可以是汉语拼音，也可以用英语标明。本程序中为简化姓名，仅用三个英文字母代替，如查找一个叫A B A的人，只要键入A B A↙即可。

性别——以W代表男性，M代表女性，查找时，若输入W↙。则可检索打印出全部男同志的档案信息。

年龄——L，如找40岁的人，则键入L-40↙即可。

工龄——G，如查找工龄12年的人，则敲入G-12↙

文化程度——H，由于文化程度各人情况不同，可以用1、2、3、4、5、6 分别代表大学、大专、中专、高中毕业、高中程度、初中以下等6种情况。如键入H—2,代表查找本单位所有具有大专程度的人的档案信息。

行政级别——z，如6级，为z—06，23级为z—23，无级别可填，则为z—99。

技术职称——K，这也可以根据使用单位实际情况，人为定义，如教授为K01，研究员K02，高工K03、副教授K04，副研K05、讲师K06、助研K07、工程师K08、助教K09、助工K10……，技师K13，无职称可填K99。

政治面貌——I，如党员为P—D，团员为P—T，预备党员为P—Y，群众为P—Q。

健康状况——T，健康、一般、较差分别填T—1、T—2、T—3。

总之，干部档案管理程序，必需事先编制一个详细的代号明显表，规定编号的定义及使用编号的注意事项。

数据区中档案的信息，必须按规则填写，例如上面提及的，每项之间以一个空格为分隔。

(3) 本程序10个人的档案资料：

```
]RUN
B1 ABA W L-40 G-13 H-2 Z-22 K-06 P-D T-2
B2 SSJ M L-47 G-21 H-1 Z-20 K-02 P-Y T-1
B3 QIQ W L-64 G-36 H-3 Z-14 K-13 P-T T-3
```

```
B4 PPI M L-44 G-16 H-1 Z-19 K-08 P-Q T-3
B5 RHI M L-27 G-10 H-3 Z-23 K-99 P-T T-1
B6 RRC W L-37 G-16 H-1 Z-20 K-09 P-D T-1
B7 STJ M L-49 G-28 H-2 Z-16 K-13 P-Y T-2
B8 EQR M L-55 G-30 H-3 Z-15 K-99 P-D T-3
B9 ABA M L-40 G-12 H-2 Z-23 K-06 P-D T-2
B10 QIQ W L-64 G-36 H-3 Z-14 K-12 P-T T-3
-----------------------------------------
```

程序H88查找方法及实例：

（1）按序号查找：

如查找编号为5的人的档案，则敲入B 5↙

结果为：

B 5 NO：5

* B5RHI M L-27G-10H-3 Z-23K-99P-T T-1*

即该同志编号为5，存贮在数据区的第5个位置，他叫RHI，女同志，27岁，工龄12年，中专文化程度，行政23级，无技术职称，共青团员，身体健康。

（2）按姓名查找：

如查找一个叫EQ R的人，键入EQ R↙

结果为：

EQR NO：8

* B8EQR M L-55G-30H-3Z-15K-99P-DT-3*

若查找名叫A B A的人，键入A B A↙

结果为：

A B A NO：1

* B1 ABA W L-40G-13H-2Z-22K-06P-D

T-2*

ABA NO：9

B9ABA M L-40 G-12 H-2 Z-23 K-06 P-D T-2

因为本程序中共10个人，ABA同名同姓的人有两个，故有以上结果。

注意，由于程序H88查找方式多样，所以在按姓名查找时，可以输入以下情况中的任一种，都可查出：

输入全名（姓和名）；

输入姓；

输入名字；

输入姓名中相邻的某几个特征字符（见下例）。

（3）按名中相邻的某几个特征字符：

如键入IQ↙

运行结果是：

IQ NO：3

B3 QIQ WL-64 G-36 H-3 Z-14 K-13 P-T T-3

IQ NO：10

B10 QIQ WL-64 G-36 H-3 Z-14 K-12 P-T T-3

这两个人的情况，几乎一样，唯一区别是在技术职称上不同，前者为K-13是技师，后者为K-12，为技术员（如果原来定义K-12为技术员的话）。

（4）按性别查找

如键入W↙，这是查出所有男同志的档案，程序H88共

有 4 人，序号为1,3,6,10。

如键入M↙，则查找打印出全部女职工的信息。

（5）按政治面貌查找：

如键入P－Y↙，是查预备党员的人事情况的，程序H88中仅有二人，编号为2，7。

（6）符合多个条件的查找：

如查找无技术职称的党员，键入K-99 P-D

程序H88只有一人，结果如下：

K-99 P-D NO：8

* D8 EQR M L-55 G-30 H-3 Z-15 K-99 P-D T-3 *

这里请特别注意一下，程序H88在查找同时满足两个或两个以上条件时，必须是相邻的两个或两个以上的项，否则是找不到的。

所谓相邻的项，在上面的例子中K-99 P—D是相邻的项，而L-55和H-3就不是相邻的项，前者可以查找出来，而后者则不行。查找不到的以“————……——”为标志。

又如查找男同志年龄为37岁，工龄为16年，大学文化程度，这四个同时必须满足的条件时，则有可能查找出，因为性别、年龄、工龄、文化程度四项是相连在一起的（若数据区中有满足上述四个条件的人，则一定能查出）。

如送入W L-37 G-16 H-1↙ 则打印以下信息：

W L-37 G-16 H-1 NO：6

* B6 RRC W L-37 G-16 H-1 Z-20 K09 P-D T-1 *

（7）按文化程度查找：

如查找所有大专文化程度的人，则键入H-2↙即可。

运行结果有3人。

程序的查找实例远不止上述几种，仅举数例，可见一般。总之，程序H88查找的灵活性是很大的，使用时也是非常方便的，读者可以用多种方式运行程序，从中体会程序的通用、灵活、方便、易懂的特点。

现在，我们来谈一下找同时满足几个条件，而各项又不是紧连在一起的检索问题。

方法是在上面介绍的程序H88基础上加改如下语句：

```
52  INPUT "X="; X
53  ON  X  GOSUB  55, 300,170
54  GOTO  52
160 RETURN
70  END
```

RON↙后，程序45句，打印全部职工档案信息。

然后转至52句，若此时输入1，执行55句，同程序H88运行方法。

若输入3，则程序转向170句结束。

而当输入X = 2时，则转向300句开始的一段子程序，即：

```
300 INPUT "L="; L$,"H=";H$,"K=";K$
305 FOR  K=1 TO  10
310 IF L$=MID$ (A$(K).10,4)AND
     H$=MID$ (A$(K).20,3)AND
     K$=MID$(A$ (K).29,3)THEN LPPI
 NT A$(K)
315 NEXT  K
```

```
317 PRINT "+*+*+*+*+*+*+*+*
     +*"
320 RETURN
```

从300句到320句，是为查找同时满足年龄、文化程度、技术职称这三个条件而设置的，读者不难分析它的原理。

实例如下：

若顺序键入：L-64↙，H-3↙，K13↙，则有以下结果：

B3 QIQ W L-64 G-36 H-3 Z-14 K13 P-T T-3

若顺序键入：L-44↙，H-1↙，K08↙，则有：

B4 PPI M L-44 G-16 H-1 Z-19 K08 P-Q T-3。

下面给出程序H88清单和部份运行结果：

```
]LIST

1  REM H88
2  REM   情报检索程序2
5  REM  SEARCHING PROGRAM
10  GOSUB 500
15  PRINT
20  DIM A$(10)
30  FOR J = 1 TO 10
40  READ A$(J)
45  PRINT A$(J)
50  NEXT J
52  PRINT "-----------------------------------
-------"
53  PRINT
55  INPUT B$
57  IF B$ = "#" THEN 160
```

```
58 N = 0
60  FOR J = 1 TO 10
70 N = N + 1
80  FOR I = 1 TO  LEN (A$(J)) -  LEN (B$) +
1
90  IF B$ < > MID$ (A$(J),I, LEN (B$)) THE
N 120
100  PRINT B$;" ";"No:";" ";N
110  PRINT "*";A$(J);"*"
120  NEXT I
130  NEXT J
140  PRINT "----------------------------------
--------"
150  GOTO 55
160  END

200  DATA "B1 ABA W L-40 G-13 H-2 Z-22 K-06
P-D T-2"
210  DATA "B2 SSJ M L-47 G-21 H-1 Z-20 K-02
P-Y T-1"
220  DATA "B3 QIQ W L-64 G-36 H-3 Z-14 K-13
P-T T-3"
230  DATA "B4 PPI M L-44 G-16 H-1 Z-19 K-08
P-Q T-3"
240  DATA "B5 RHI M L-27 G-10 H-3 Z-23 K-99
P-T T-1"
250  DATA "B6 RRC W L-37 G-16 H-1 Z-20 K-09
P-D T-1"
260  DATA "B7 STJ M L-49 G-28 H-2 Z-16 K-13
P-Y T-2"
270  DATA "B8 EQR M L-55 G-30 H-3 Z-15 K-99
```

```
P-D T-3"
280  DATA "B9 ABA M L-40 G-12 H-2 Z-23 K-06
P-D T-2"
290  DATA "B10 QIQ W L-64 G-36 H-3 Z-14 K-12
 P-T T-3"
500  PRINT "本程序有下列检索方法"
510  PRINT "1.按序号查找"
520  PRINT "2.按姓名查找"
530  PRINT "3.按姓名中特征字查找"
540  PRINT "4.按性别查找"
550  PRINT "5.按政治面貌查找"
560  PRINT "6.按几个条件查找"
570  PRINT "7.按文化程度查找"
580  PRINT "8.按个别字符查找"
590  RETURN

]RUN
本程序有下列检索方法
1.按序号查找
2.按姓名查找
3.按姓名中特征字查找
4.按性别查找
5.按政治面貌查找
6.按几个条件查找
7.按文化程度查找
8.按个别字符查找

B1 ABA W L-40 G-13 H-2 Z-22 K-06 P-D T-2
B2 SSJ M L-47 G-21 H-1 Z-20 K-02 P-Y T-1
B3 QIQ W L-64 G-36 H-3 Z-14 K-13 P-T T-3
B4 PPI M L-44 G-16 H-1 Z-19 K-08 P-Q T-3
B5 RHI M L-27 G-10 H-3 Z-23 K-99 P-T T-1
```

```
B6 RRC W L-37 G-16 H-1 Z-20 K-09 P-D T-1
B7 STJ M L-49 G-28 H-2 Z-16 K-13 P-Y T-2
B8 EQR M L-55 G-30 H-3 Z-15 K-99 P-D T-3
B9 ABA M L-40 G-12 H-2 Z-23 K-06 P-D T-2
B10 QIQ W L-64 G-36 H-3 Z-14 K-12 P-T T-3
----------------------------------------

?B5
B5 No: 5
*B5 RHI M L-27 G-10 H-3 Z-23 K-99 P-T T-1*
----------------------------------------
?EQR
EQR No: 8
*B8 EQR M L-55 G-30 H-3 Z-15 K-99 P-D T-3*
----------------------------------------
?ABA
ABA No: 1
*B1 ABA W L-40 G-13 H-2 Z-22 K-06 P-D T-2*
ABA No: 9
*B9 ABA M L-40 G-12 H-2 Z-23 K-06 P-D T-2*
----------------------------------------
?IQ
IQ No: 3
*B3 QIQ W L-64 G-36 H-3 Z-14 K-13 P-T T-3*
IQ No: 10
*B10 QIQ W L-64 G-36 H-3 Z-14 K-12 P-T T-3*
----------------------------------------

?W
W No: 1
*B1 ABA W L-40 G-13 H-2 Z-22 K-06 P-D T-2*
W No: 3
```

```
*B3 QIQ W L-64 G-36 H-3 Z-14 K-13 P-T T-3*
W No: 6
*B6 RRC W L-37 G-16 H-1 Z-20 K-09 P-D T-1*
W No: 10
*B10 QIQ W L-64 G-36 H-3 Z-14 K-12 P-T T-3*
-----------------------------------------
?P-Y
P-Y No: 2
*B2 SSJ M L-47 G-21 H-1 Z-20 K-02 P-Y T-1*
P-Y No: 7
*B7 STJ M L-49 G-28 H-2 Z-16 K-13 P-Y T-2*
-----------------------------------------
?K-99 P-D
K-99 P-D No: 8
*B8 EQR M L-55 G-30 H-3 Z-15 K-99 P-D T-3*
-----------------------------------------
?W L-37 G-16
W L-37 G-16 No: 6
*B6 RRC W L-37 G-16 H-1 Z-20 K-09 P-D T-1*
-----------------------------------------
?H-2
H-2 No: 1
*B1 ABA W L-40 G-13 H-2 Z-22 K-06 P-D T-2*
H-2 No: 7
*B7 STJ M L-49 G-28 H-2 Z-16 K-13 P-Y T-2*
H-2 No: 9
*B9 ABA M L-40 G-12 H-2 Z-23 K-06 P-D T-2*
-----------------------------------------
?M
M No: 2
*B2 SSJ M L-47 G-21 H-1 Z-20 K-02 P-Y T-1*
```

```
M No: 4
*B4 PPI M L-44 G-16 H-1 Z-19 K-08 P-Q T-3*
M No: 5
*B5 RHI M L-27 G-10 H-3 Z-23 K-99 P-T T-1*
M No: 7
*B7 STJ M L-49 G-28 H-2 Z-16 K-13 P-Y T-2*
M No: 8
*B8 EQR M L-55 G-30 H-3 Z-15 K-99 P-D T-3*
M No: 9
*B9 ABA M L-40 G-12 H-2 Z-23 K-06 P-D T-2*
----------------------------------------
?#
```

下面给出一个用汉字存贮的人事档案程序H89，检索方法同上，但可以直接用汉字检索，从而使程序更为实用。

```
]LIST

1  REM H89
2  REM 情报检索程序1
4  GOSUB 500
5  PRINT
10  REM   SEARCHING PROGRAM
20  DIM A$(50)
25  ONERR  GOTO 55
30  FOR J = 1 TO 50
40  READ A$(J)
50  NEXT J
51  PRINT "--------------------------------
-------"
52  PRINT
55  INPUT B$
```

```
57  IF B$ = "#" THEN 160
58 N = 0
60  FOR J = 1 TO 12
70 N = N + 1
80  FOR I = 1 TO  LEN (A$(J)) -  LEN (B$) +
1
90  IF B$ < > MID$ (A$(J),I, LEN (B$)) THE
N 120
100  PRINT B$;" ";"No:";" ";N
110  PRINT "*";A$(J);"*"
120  NEXT I
130  NEXT J
135  RESTORE
140  PRINT "----------------------------------
----------------------------"
150  GOTO 55
160  END
200  DATA "3421 朱小民 男 40岁 工龄13 本科 8
9.00元 工程师 团员 健康"
210  DATA "3422 李  民 男 20岁 工龄4 初中 45
.60元 工  人 团员 健康"
220  DATA  "3423 吴小敏 女 30岁 工龄8 高中 7
6.50元 工  人 团员 一般"
230  DATA  "3424 邹丽莉 女 22岁 工龄2 大专 6
0.50元 技术员 群众 一般"
240  DATA "3425 张  立 男 46岁 工龄20 大专 9
7.00元 工程师 群众 尚可"
250  DATA "3426 安金立 男 55岁 工龄35 本科 1
22.0元 高  工 党员 一般"
260  DATA "3427 俞大为 男 55岁 工龄40 高小 1
34.0元 技  师 党员 一般"
```

```
270 DATA "3428 王元苑 女 35岁 工龄15 高中
87.00元 工  人 党员 健康"
280 DATA "3429 常文林 男 42岁 工龄18 高中 1
03.0元 技  师 党员 健康"
290 DATA "3430 胡文敬 男 53岁 工龄32 本科 1
60.0元 教  授 党员 健康"
300 DATA "3431 方学琴 女 49岁 工龄27 本科
134.0元 副教授 党员 健康"
310 DATA "3432 林平平 女 45岁 工龄22 本科
122.0元 讲  师 群众 健康"
500 PRINT "本程序有下列检索方法"
510 PRINT "1.按序号查找"
520 PRINT "2.按姓名查找"
530 PRINT "3.按姓名中特征字查找"
540 PRINT "4.按性别查找"
550 PRINT "5.按政治面貌查找"
560 PRINT "6.按几个条件查找"
570 PRINT "7.按文化程度查找"
580 PRINT "8.按个别字符查找"
590 RETURN

]RUN
本程序有下列检索方法
1.按序号查找
2.按姓名查找
3.按姓名中特征字查找
4.按性别查找
5.按政治面貌查找
6.按几个条件查找
7.按文化程度查找
8.按个别字符查找
```

?女
女 No: 3
3423 吴小敏 女 30岁 工龄8 高中 76.50元 工人 团员 一般
女 No: 4
3424 邹丽莉 女 22岁 工龄2 大专 60.50元 技术员 群众 一般
女 No: 8
3428 王元苑 女 35岁 工龄15 高中 87.00元 工人 党员 健康
女 No: 11
3431 方学琴 女 49岁 工龄27 本科 134.0元 副教授 党员 健康
女 No: 12
3432 林平平 女 45岁 工龄22 本科 122.0元 讲师 群众 健康

?男
男 No: 1
3421 朱小民 男 40岁 工龄13 本科 89.00元 工程师 团员 健康
男 No: 2
3422 李　民 男 20岁 工龄4 初中 45.60元 工人 团员 健康
男 No: 5
3425 张　立 男 46岁 工龄20 大专 97.00元 工程师 群众 尚可
男 No: 6
3426 安金立 男 55岁 工龄35 本科 122.0元 高工 党员 一般

男 No: 7
*3427 俞大为 男 55岁 工龄40 高小 134.0元 技
师 党员 一般*
男 No: 9
*3429 常文林 男 42岁 工龄18 高中 103.0元 技
师 党员 健康*
男 No: 10
*3430 胡文敬 男 53岁 工龄32 本科 160.0元 教
授 党员 健康*

?122
122 No: 6
*3426 安金立 男 55岁 工龄35 本科 122.0元 高
工 党员 一般*
122 No: 12
*3432 林平平 女 45岁 工龄22 本科 122.0元 讲
师 群众 健康*

?高
高 No: 3
*3423 吴小敏 女 30岁 工龄8 高中 76.50元 工
人 团员 一般*
高 No: 6
*3426 安金立 男 55岁 工龄35 本科 122.0元 高
工 党员 一般*
高 No: 7
*3427 俞大为 男 55岁 工龄40 高小 134.0元 技
师 党员 一般*

高 No: 8
*3428 王元苑 女 35岁 工龄15 高中 87.00元 工
人 党员 健康*
高 No: 9
*3429 常文林 男 42岁 工龄18 高中 103.0元 技
师 党员 健康*

?员
员 No: 1
*3421 朱小民 男 40岁 工龄13 本科 89.00元 工
程师 团员 健康*
员 No: 2
*3422 李 民 男 20岁 工龄4 初中 45.60元 工
人 团员 健康*
员 No: 3
*3423 吴小敏 女 30岁 工龄8 高中 76.50元 工
人 团员 一般*
员 No: 4
*3424 邹丽莉 女 22岁 工龄2 大专 60.50元 技术
员 群众 一般*
员 No: 6
*3426 安金立 男 55岁 工龄35 本科 122.0元 高
工 党员 一般*
员 No: 7
*3427 俞大为 男 55岁 工龄40 高小 134.0元 技
师 党员 一般*
员 No: 8
*3428 王元苑 女 35岁 工龄15 高中 87.00元 工
人 党员 健康*
员 No: 9

```
*3429 常文林 男 42岁 工龄18 高中 103.0元 技
 师 党员 健康*
员 No: 10
*3430 胡文敬 男 53岁 工龄32 本科 160.0元 教
 授 党员 健康*
员 No: 11
*3431 方学琴 女 49岁 工龄27 本科 134.0元 副
教授 党员 健康*
--------------------------------------------------

----------------

?#
```

最后给出一个英文文献查找的实例，见程序H90：

```
0  REM H90
1  GOSUB 500
2  REM  情报检索程序
5  REM  SEARCHING PROGRAM
20  DIM A$(10)
30  FOR J = 1 TO 10
40  READ A$(J)
50  NEXT J
52  PRINT "--------------------------------
--------------------------------------------------

---"
53  PRINT
55  INPUT B$
57  IF B$ = "#" THEN 160
58 N = 0
60  FOR J = 1 TO 10
70 N = N + 1
80  FOR I = 1 TO  LEN (A$(J)) -  LEN (B$) +
```

```
1
90 IF B$ < > MID$ (A$(J),I, LEN (B$)) THE
N 120
100 PRINT B$;" ";"No:";" ";N
110 PRINT "*";A$(J);"*"
120 NEXT I
130 NEXT J
140 PRINT "...............................
........................................
...."
150 GOTO 55
160 END

200 DATA "*Zhan DaoJiang,Zou Jinshang:The P
robable Maximum of Rainstorms and Flood,Hydr
oelectricity Press,Beijing,PP 424,1983."
210 DATA "Zou Jinshang,Liu Changshang,Liu W
enbao:The Foundation of Atmospheric Physics,
Meteorological Press,Beijing,PP 365,1982."
220 DATA "Zou Jinshang,Zhang Youlian,Yang Z
hongqiu:Climatic features in the free atmosp
here over Nanjing,ACTA meteor.Sinica,31(4),3
28-338,1962."
230 DATA "Zou Jinshang,Yang Zhongqiu,Zhang
YOULIAN:Specific humidity and water vapour c
ontent in free atmosphere over Nanjing,J.of
Nanjing Uni.,(Meteor.) No.2,39-54,1962."
240 DATA "Zou Jinshang,Zhang Youlian:Some f
avourable weather conditions for stimulating
 rains over Nanjing Region,J.of Nanjing Uni.
,(meteor.) No.2,85-97,1962."
250 DATA "Zou Jinshang,Mo Tianlin and Xu Sh
aozhu:Some aspects of the distubution of gia
nt hygroscopic nuclei over the region of the
 lower reaches of Yangtze in summer,J.of Nan
jing Uni.,8(1),148,165,1964."
```

```
260  DATA "Zou Jinshang:A method of estimati
on PMP,J.of Nanjing Uni.,No.1,113-118,1977."
270  DATA "Zou Jinshang:('756'group of Nanji
ng Uni.) Estimation in the valley of Tou-tou
 River Meteorological Science and Technology
,13-33,No.5,1977."
280  DATA "Zou Jinshang:Some methods of comp
uting water vapour content in the atmosphere
,Meteorological Science and Technology,33-36
,1978."
290  DATA "Zou Jinshang:Advance of study on
PMP Estimation,Natural Journal,3(11),815-816
,1980."
500  PRINT "本程序以查找英文文献为模式,介绍
了一种简短易懂,操作方便的多用途检索程序."
510  PRINT "程序运行后,输入文献中的任意关键
字(个别字符),即可查找存储的信息,如无此文献以
.....标志找不到,并返回主控再找,停止查找按#和
RETURN"
600  RETURN

]RUN
本程序以查找英文文献为模式,介绍了一种简短易
懂,操作方便的多用途检索程序.
程序运行后,输入文献中的任意关键字(个别字符),
即可查找存储的信息,如无此文献以.....标志找不
到,并返回主控再找,停止查找按#和RETURN
--------------------------------------------
-------------------------------------

?1962
1962 No: 3
*Zou Jinshang,Zhang Youlian,Yang Zhongqiu:Cl
imatic features in the free atmosphere over
Nanjing,ACTA meteor.Sinica,31(4),328-338,196
2.*
1962 No: 4
*Zou Jinshang,Yang Zhongqiu,Zhang YOULIAN:Sp
ecific humidity and water vapour content in
```

free atmosphere over Nanjing,J.of Nanjing Uni.,(Meteor.) No.2,39-54,1962.*
1962 No: 5
Zou Jinshang,Zhang Youlian:Some favourable weather conditions for stimulating rains over Nanjing Region,J.of Nanjing Uni.,(meteor.) No.2,85-97,1962.

....

?Maximum
Maximum No: 1
**Zhan Daojiang,Zou Jinshang:The Probable Maximum of Rainstorms and Flood,Hydroelectricity Press,Beijing,PP 424,1983.*

....

?85-97
85-97 No: 5
Zou Jinshang,Zhang Youlian:Some favourable weather conditions for stimulating rains over Nanjing Region,J.of Nanjing Uni.,(meteor.) No.2,85-97,1962.

....

?PP 424
PP 424 No: 1
**Zhan Daojiang,Zou Jinshang:The Probable Maximum of Rainstorms and Flood,Hydroelectricity Press,Beijing,PP 424,1983.*

....

?Meteorological
Meteorological No: 2
Zou Jinshang,Liu Changshang,Liu Wenbao:The Foundation of Atmospheric Physics,Meteorological Press,Beijing,PP 365,1982.
Meteorological No: 8
*Zou Jinshang:('756'group of Nanjing Uni.) E

stimation in the valley of Tou-tou River Meteorological Science and Technology, 13-33, No. 5, 1977.*
Meteorological No: 9
Zou Jinshang:Some methods of computing water vapour content in the atmosphere, Meteorological Science and Technology, 33-36, 1978.
....

?atmosphere
atmosphere No: 3
Zou Jinshang, Zhang Youlian, Yang Zhongqiu:Climatic features in the free atmosphere over Nanjing, ACTA meteor. Sinica, 31(4), 328-338, 1962.
atmosphere No: 4
Zou Jinshang, Yang Zhongqiu, Zhang YOULIAN:Specific humidity and water vapour content in free atmosphere over Nanjing, J. of Nanjing Uni., (Meteor.) No. 2, 39-54, 1962.
atmosphere No: 9
Zou Jinshang:Some methods of computing water vapour content in the atmosphere, Meteorological Science and Technology, 33-36, 1978.
....

?424, 19
?EXTRA IGNORED
424 No: 1
**Zhan Daojiang, Zou Jinshang:The Probable Maximum of Rainstorms and Flood, Hydroelectricity Press, Beijing, PP 424, 1983.*
....

?Zou
Zou No: 1
**Zhan Daojiang, Zou Jinshang:The Probable Maximum of Rainstorms and Flood, Hydroelectricity Press, Beijing, PP 424, 1983.*

Zou No: 2
Zou Jinshang,Liu Changshang,Liu Wenbao:The Foundation of Atmospheric Physics,Meteorological Press,Beijing,PP 365,1982.
Zou No: 3
Zou Jinshang,Zhang Youlian,Yang Zhongqiu:Climatic features in the free atmosphere over Nanjing,ACTA meteor.Sinica,31(4),328-338,1962.
Zou No: 4
Zou Jinshang,Yang Zhongqiu,Zhang YOULIAN:Specific humidity and water vapour content in free atmosphere over Nanjing,J.of Nanjing Uni.,(Meteor.) No.2,39-54,1962.
Zou No: 5
Zou Jinshang,Zhang Youlian:Some favourable weather conditions for stimulating rains over Nanjing Region,J.of Nanjing Uni.,(meteor.) No.2,85-97,1962.
Zou No: 6
Zou Jinshang,Mo Tianlin and Xu Shaozhu:Some aspects of the distubution of giant hygroscopic nuclei over the region of the lower reaches of Yangtze in summer,J.of Nanjing Uni., 8(1),148,165,1964.
Zou No: 7
Zou Jinshang:A method of estimation PMP,J.of Nanjing Uni.,No.1,113-118,1977.
Zou No: 8
Zou Jinshang:('756'group of Nanjing Uni.) Estimation in the valley of Tou-tou River Meteorological Science and Technology,13-33,No.5,1977.
Zou No: 9
Zou Jinshang:Some methods of computing water vapour content in the atmosphere,Meteorological Science and Technology,33-36,1978.
Zou No: 10
*Zou Jinshang:Advance of study on PMP Estima

tion,Natural Journal,3(11),815-816,1980.*
....
....
?#

九、图 形 绘 制

1. 电子日历钟

下面是一个电子日历钟程序（见H91），只要在程序运行后输入当时的月份、日期、星期和时间（时间以整小时输入），即可在屏幕上显示（如图9.1）的一个时钟，它显示醒目，走时准确，毫不逊色于一个真正的电子钟。

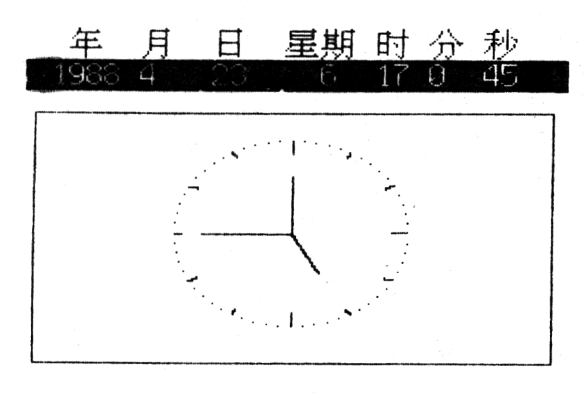


图 9.1

由于图形和汉字信息全部在高分辨率第2页画面作出，故打印十分方便，按CTRL-RESET后，键入：

PR#1↙

POKE1913，2↙

PRINT CHR$(17)↙

```
]LIST

5  REM H91
10  REM 电子日历钟
20  DIM M(12)
25  FOR I = 1 TO 12: READ M(I): NEXT I
30  HGR2 : HOME
35  VTAB 1: HTAB 1: INPUT "请输入月,日,星期,
时:";T,D,W,H
36  IF T < 1 OR T > 12 THEN 30
37  IF D < 1 OR D > M(T) OR W < 0 OR W > 6 O
R H < 0 OR H > 23 THEN 30
40 K1 = 0:K2 = 0
50 N = 3.1415926 * 2 - 3.1415926 * 2 / 60
60 ND = 3.1415926 * 2 / 60
70  HCOLOR= 3
80  HPLOT 25,45 TO 255,45 TO 255,179 TO 25,1
79 TO 25,45
90  FOR S = 0 TO N STEP ND
100 X = 52 *  SIN (S) + 140:Y =  - 50 *  COS
 (S) + 110
110  HPLOT X,Y
120  NEXT S
130  FOR S = 0 TO N STEP ND * 5
137  IF S = 45 * ND THEN Q = 0.85
140 X = 52 *  SIN (S) + 140:Y =  - 50 *  COS
 (S) + 110
145 Q = 0.93: IF S = 0 OR S = 15 * ND OR S =
 30 * ND THEN Q = 0.85
150 XX = (X - 140) * Q + 140
160 YY = (Y - 110) * Q + 110
```

```
170  HPLOT X,Y TO XX,YY
180  NEXT S
270  FOR I1 = T TO 12
280  FOR I2 = D TO M(I1)
290  FOR I3 = H TO 23
300  FOR M = 0 TO N STEP ND
310 X2 = 25 *  SIN (0.5236 * I3 + M / 12) +
140
320 Y2 =  - 25 *  COS (0.5236 * I3 + M / 12)
 + 110
330 X = 32 *  SIN (M) + 140:Y =  - 31 *  COS
 (M) + 110
340  FOR S = 0 TO N STEP ND
350 X1 = 40 *  SIN (S) + 140:Y1 =  - 40 *  C
OS (S) + 110
360  HCOLOR= 3
370  HPLOT 140,110 TO X2,Y2
380  HPLOT 140,110 TO X,Y
390  HPLOT 140,110 TO X1,Y1
400  FOR I = 1 TO 522: NEXT
410  HCOLOR= 4
420  HPLOT 140,110 TO X1,Y1
430 K1 = K1 + 1: IF K1 = 60 THEN K1 = 0
440  VTAB 1: PRINT  TAB( 6);"年"; TAB( 10);"
月"; TAB( 14);"日"; TAB( 18);"星期"; TAB( 23
);"时"; TAB( 26);"分"; TAB( 29);"秒"
460  VTAB 2: HTAB 2
470  INVERSE
480  PRINT  TAB( 5);"1988"; TAB( 10);I1; TAB
( 14);I2; TAB( 20);W; TAB( 23);I3; TAB( 26);
K2; TAB( 29);K1;"    "
```

```
490  NORMAL
500  NEXT S
510  HCOLOR= 4
520  HPLOT 140,110 TO X2,Y2
530  HPLOT 140,110 TO X,Y
540 K2 = K2 + 1
550  IF K2 = 60 THEN K2 = 0: PRINT "": PRINT
 "": PRINT "": REM CTRL-G
560  NEXT M
570  NEXT I3
580 H = 0
590 W = W + 1
600  IF W = 7 THEN W = 0
610  NEXT I2
620 D = 1
```

程序简要说明：

· 20，25，650三句：存贮月份(M（12）)，注意润年，如1988年 2月为29天。

· 30—37句：输入当时的月份、日期、星期和时间。星期以0,1,2,……，6，其中 0代表星期日，时间以小时输入为单位，计时从 0分 0秒开始。36—37句为防止输入错误而设置的控制语句。

· 40—180句:画电子钟圆盘,其中80句画一个方框,90—120句画60个小点，130—180句画12根短线。

· 270—630句为一个大循环，循环变量为I1，计月份。

· 280—610句，计日期，循环控制变量为I2。

· 290—570句，计算小时，用I3做循环控制变量。

·300—560句，控制分、秒循环。K_2，K_1为分、秒计数器。

·370—390句，画时、分、秒针。

·调正400句的循环终值，可使电子钟走时准确。

·440句，打印中文年月日星期时分钟的表头。

·480句，打印年月日星期时分秒的数字，它们随着时钟的走动而变化。

·430句，控制秒的计时，计满60秒K 1复零。540—550句，控制分的计时，计满60分K 2复零。590—600句，控制星期，计满6天W复零。I_3循环完成，小时H复零。I_2循环完成，日期D从1号开始。

2. 高分辨率图形的向量作图法

下面是计算机打印出来的一张电子线路图（见图9.2）。绘制电路图形不是一件困难的工作，它可以用高分辨图象方式下的画图语句HPLOT完成。但是，要在电路图的适当地方，加注元件符号就不是一个简单的事情，因为许多微机都存在高分辨作图和文本方式互不兼容的问题，这就是说，可

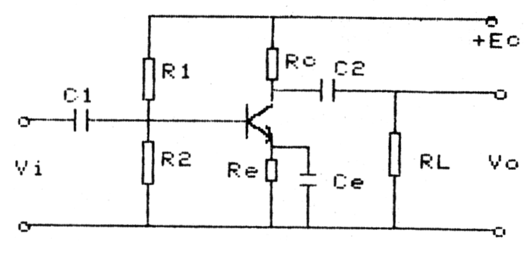


图 9.2

以绘图但不能写字，特别是不能在图象的任意位置写字。

为了解决作图和写字的矛盾，真正做到图文并茂，使图象信息更为丰富，画面更富有情趣，我们这里介绍一种新的画图方法——向量作图法。

向量作图法，又称矢量作图法，或者称为图形法，造型表方法。这种方法的基本思想是：把图的轮廓线分解成许多基本矢量的集合，把绘图时笔划移动的过程记录下来，做成图形表（造型表），将造型表放入内存中某个特定区域，它就是绘图的依据，然后在程序中用特定的语句和专门的命令，调用这个造型表，就可以在屏幕上将图形显示出来，同时，利用绘图命令，可以将调出来的图形进行旋转、放大。

上述思想是不难理解的，因为任何复杂的图形都可以分解成若干个基本图形，而每一个基本图形都可以看成是很多短线、点线的集合。因此，只要控制这些短线或点的走向，就可以绘出各种图形。例如，一幅风景画可以由山脉、河流、树木、花草、房屋、小桥组成，而山脉、河流、花草等都是由点点线线构成的，改变它们比例的大小和相对位置，就可以构成不同的风景画面。如果再加上有关作图语句和命令还可以使图形活动起来，从而形成美丽多变妙趣横生的动画来。

下面分五个问题介绍高分辨率图形造型表。

(1) 造型表生成

假定我们希望画出如图 9.3 所示的图形，可先拿一张方格纸来具体画出，然后用图形向量依次记录笔的移动过程，当笔向什么方向移动，就画一个向那个方向的箭头；移出这一格子时如需画点，则在箭根上画一个圆点。图 9.3 中的 6

个点，可用图9.4中表示的a—g 7个步骤完成。

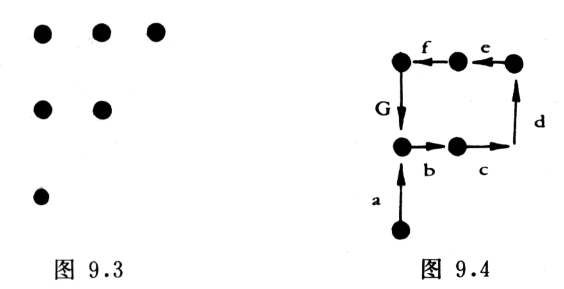


图 9.3　　　　图 9.4

图9.4中a表示画一个点，然后向上移动一个坐标；b表示画一个点，然后向右移动一个坐标；c与b相同。但d则表示不画点，仅上移一个坐标。

至于从那一点开始，那是没有关系的，但一定先选定一个笔的起始位置，本例中是从a矢量开始画。

以上是一个简单图形的描绘过程，但是仅有这张向量图还不行，如何区别向什么方向移动，画还是不画呢？显然必须用一定的特征代码来表示，这样机器才好识别。

例如向上、向右、向下、向左四个移动方向可以用特性码00，01，10，11分别表示：即

移动方向	特征代码
↑ 向上	00
→ 向右	01
↓ 向下	10
← 向左	11

但是仅有二个代码还不行，因为它们只能表示移动方向，还不能说明画或者不画的情况，为此必须再选用一个代码，以区分画与不画两种情况，例如，我们约定1表示画，0表示

不画。这样，四个移动方向，画与不画共8种情况就可以表示出来，即：

↑	向上不画	000
→	向右不画	001
↓	向下不画	010
←	向左不画	011
↥	向上画出	100
•→	向右画出	101
↧	向下画出	110
←•	向左画出	111

有了上述特性代码，图9.4中a—g对应的描绘向量段应分别办：

a：100

b：101

c：101

d：000(或00)

e：111

f：111

g：110

为了将上述各个描绘向量所对应的特征代码存贮在内存区中，必须按一定的规则顺序存放。由于8位机每个字节可

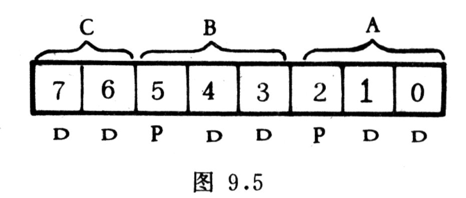


图 9.5

以存放8位二进制代码，因此，每个字节可以存贮两个描绘矢量对应的特征代码。

每个字节分成三段，如图9.5所示：

其中最低三位为A段；中间三位为B段；最高二位为C段。A、B两段中三位数的低二位及C段中的二位数称做D位；A、B两段中的高位称为P位。对于D和P做如下规定：

$$P=\begin{cases}0 & \text{不画}\\1 & \text{画点}\end{cases}$$

$$DD=\begin{cases}00 & \uparrow\\01 & \rightarrow\\10 & \downarrow\\11 & \leftarrow\end{cases}$$

这样，就和我们先前约定的描绘每个矢量的特性码相一致。

为了将图9.4中描绘向量的特性码填入图9.5所示的字节中，应按下述规定处理：

- 每个描绘矢量按三位码顺序从A段至C段填写。
- 对每个描绘矢量来说，如果C＝00，则C段不起作用，即不用C段；如果B＝000，则仅A段起作用。
- 当一个图形的所有描绘矢量填完后，必须填满一个全零字节，作为结束表示。
- 在表示“上移不绘出”这个描绘矢量时，高端必须非零，否则不会产生任何移动。即“上移不绘”这个矢量在B段表示时，C段必须为0。

这样，图9.4的情况应按图9.6格式存放。

C段	B段	A段
0 0	101	100
0 0	000	101
0 0	111	000
0 0	110	111

图 9.6

以上介绍的仅仅是一个造型，即图9.3中6个点的图形。如果再加上这个造型的结束标志，将全0填满一个字节，则共占5个字节。

应该指出的是，每个字节中8位二进制代码书写十分不便，常常改用十进制，八进制或十六进制填写。如二进制码“00101111”改写成其它进制形式为：

二进制码	二进制码	二进制码
00101 111	00101 111	00101 111
十进制码	八进制码	十六进制码
47	5，7	2 F

（2）造型表存贮

怎样将描绘向量的图形数据（即特性码）检索出来，以便以后用绘图语句画出图象的原形来，首先要解决各个造型的存贮问题，为了方便检索，还应编制一个索引，这样就要编制一个造型表，它由许多图形数据和索引信息组成。

造型表中前面部份安排一个索引挡，它指明造型表中有多少个造型，每一个造型的存贮地址，接下去存放每一个造型矢量的代码，每一个造型用全0字节结尾。而其它不同的造型一个接着一个地装在表中。

现在，我们从一个具体例子出发，看造型表是如何编制的。

例如，有五个造型数据，它们分别为：

45，40，32，5，168，174，21，37，0

45，40，32，5，40，45，21，37，0

45，5，40，5，40，45，21，37，0

45，45，45，45，45，37，0，37，0

9，5，32，5，40，168，21，21，173，37，0

假定索引挡首地址放在＄300(768)中，则造型表的结构如表9.1。

表 9.1

	地　址	存贮内容	说　明
S	＄0300	05	本图形表包含图形造型的个数。本例为5.S是索引表首址。
S＋1	＄0301	00	规定值。
S＋2 S＋3	＄0302 ＄0303	0C 00	第一个造型首地址从＄030C开始，离表头的距离是12（＄000C，低位在前，高位在后）。
S＋4 S＋5	＄0304 ＄0305	15 00	第二个造型首地址　从＄0315开始，离表头的距离21（＄0015）。
S＋6 S＋7	＄0306 ＄0307	1E 00	第三个造型首地址从＄031E开始，离表头的距离30（＄001E）。
S＋8 S＋9	＄0308 ＄0309	27 00	第四个造型首地址从＄0327开始，离表的距离39（＄0027）。
S＋10 S＋11	＄030A ＄030B	2E 00	第五个造型首地址从＄032E开始，离表头的距离46（＄002E）。
S＋12	＄030C	2D	第一个造型的第一个数据45（＄2D），其首地址是＄030C。
S＋13	＄030D	28	第一个造型的第二个数据40（＄28），地址是＄030D。
S＋14	＄030E	20	第一个造型的第三个数据32（＄20），地址是＄030E。

续表

	地　址	存贮内容	说　明
S＋15	＄030F	05	第一个造型的第四个数据5(＄05)，地址是＄030F。
S＋16	＄0310	A8	第一个造型的第五个数据168(＄A8)，地址是＄0310。
S＋17	＄0311	AE	第一个造型的第六个数据174(＄AE)，地址是＄0311。
S＋18	＄0312	15	第一个造型的第七个数据21（＄15)，地址是＄0312。
S＋19	＄0313	25	第一个造型的第八个数据37（＄25)，地址是＄0313。
S＋20	＄0314	00	第一个造型结束，应将00放在＄0314中。
S＋21 ↓ S＋29	＄0315 ↓ ＄031D	2D ⋮ 00	第二个造型的第一个数据45（＄2D)，其首地址是＄0315，末地址是＄031D。
S＋30 ↓ S＋38	＄031E ↓ ＄0326	2D ⋮ 00	第三个造型的第一个数据45(＄2D)，其首地址是＄031E，末地址是＄0326。
S＋39 ↓ S＋45	＄0327 ↓ ＄032D	2D ⋮ 00	第四个造型的第一个数据45(＄2D)，其首地址是＄0327，末地址是＄032D。
S＋46 ↓ S＋56	＄032E ↓ ＄0338	09 ⋮ 00	第五个造型的第一个数据9(＄09)，其首地址是＄032E，末地址是0338。

表9.1中的S是索引表的首地址，其它各存贮单元的地址，都是相对S而言的，是相对地址。例如，S＝0时，地址为＄0300，则S＝1时，地址就是＄0301。

索引表的首地址中，存放的是造型的个数。相对索引表首地址为1的地址内存中放的00，这个一个规定值。

从这个规定值以后，依次顺放第一、第二、……到第五个造型的首地址，每个地址用二个字节表示，地址高位在前地址低位在后，这样，五个造型共用10个存贮单元。

全部造型的地址存贮完成后，接下去依次存放每个造型的数据，一个造型的全部数据放完后，安排一个00作为结束，

再存放第二个造型、第三个造型……等等，直至全部造型数据存放完毕。

我们这个造型表的例子起始地址选为$0300，这是因为整个造型表的长度没有超过60个字节，如果造型表长度超过256个字节，则起始地址不允许选为$0300,可以选用$6000为起始地址。这就是说要把一个造型表送入机器中，首先要确定S的值,它表示造型表的起始地址放在内存中什么地方。这个地址要选择合适，否则程序运行时会把它破坏掉。

上述造型表，只为我们检索各个造型提供了依据，实际使用时还必须编制一个程序将它送入内存，然后还要使用绘图命令，把各个造型调出来从而恢复成真正的图形。

(3) 造型绘图命令

一个造型表中有若干个造型，每一个造型都是由基本矢量构成的。这些造型表在应用时，并不是一起使用的。用它绘图时必须要用专门的命令。

例如，要在屏幕的什么位置上，画第几个造型，这就要用到DRAW命令。造型表各个造型的矢量，只说明方向，并没有说明大小，必须用一个命令解决每个矢量画多大的问题，这就是SCALE命令。同时，为使图象显示更加生动活泼，还必须有能使造型旋转变化的命令ROT。为使画面产生活动的效果,还应有使已画造型消隐的命令XDRAW。

造型绘图命令有四个，它们可以把造型表中所定义的指定图形显示出来。

· SCALE

该命令的功能是设定图形向量的大小。当SCALE = 3时，每个图形向量对应屏幕上3个点；当SCALE= 0时，

表示最大的形状，每个图形向量绘成256个点。

· DRAW

绘出造型表内第几个造型，使用最近所选颜色，例如：

DRAW 1 AT 140，80

即将表中第1个造型绘于140行80列开始的地方。以它作为绘图的起笔点。

又如：DRAW 5

则将表中第5个造型绘出，坐标为（0,0）

· XDRAW

其功能和DRAW一样，但取其颜色为原来的补色，实用上它可以擦除上次已画过的图形（其参数应和DRAW一样）。

· ROT

该命令的功能是将图形转动一定角度。当SCALE＝1时，只有四个旋转值能为ROT使用（即0,16，32，48）；当SCALE＝2时，则有8个旋转值为ROT使用（0,8,16，24，32，40，48，56）；而当SCALE≥5时，可以有0到63共64种旋转角度。

（4）造型生成程序

前已指出，必须将造型表送入内存才能使用，送入的方法是用程序来实现。

例如，下面这段程序(见H92)，运行后可以得到一个动态的运动图形。DATA语句中的数据就是我们介绍的造型表中的存贮内容，不过数值都改为十进制数。20句中数据是索引挡的内容,共有5个图形定义,一个规定值0,以及5个造型的首地址。30—70句依次顺序存放1、2、3、4、5，5个造型数

据，每个DATA区放一个造型。80—100句是一个循环，I循环变量控制地址，地址中的内容放入X中，再用POKE指令存放地址和对应地址的内容。

这里都是用十进制数表示的，768就对应于十六进制数$0300，824对应于$0338。

110句的232（$E8)和233（$E9）号单元，是专门用来存放造型表首地址的。其中232单元存放造型表首地址的低位地址，233单元存放造型表首地址的高位地址。由于造型表首地址是$0300，所以有：

POKE232,0:POKE 233,3

120—190句，用循环的方式，调用5个造型，选用四种绘图语句，从而产生一个运动的画面。

改变140句的循环值，改变150，160句X后面的值，以及再加一重J循环，会使动画更加丰富，读者可以自行试验。

190句的作用，是多次重演动画，要结束按CTRL-RESET键。

程序H92在中文或西文状态下都能运行。

```
]LIST

5  REM H92
10  REM  动画程序
20  DATA  5,0,12,0,21,0,30,0,39,0,46,0
30  DATA 45,50,32,5,168,174,21,37,0
40  DATA 45,40,32,5,40,45,21,37,0
50  DATA 45,5,40,5,40,45,21,37,0
60  DATA 45,45,45,45,45,37,0
70  DATA 9,5,32,5,40,168,21,21,173,37,0
80  FOR I = 768 TO 824
```

表 9.2

工作过程	提示符	键盘输入	说明
编制造型表并存盘	]	CALL-151↙	进入监控状态
	*	0300：01 00 04 00 12 3F 24 24 2D 2D 36 36 3F 00↙	将造型表装入内存，起始地址为$0300，共14个字节，1个造型
	*	CTRL-RESET↙	返回APPLE SOFT
	]	BSAVE M，A$0300，L14	将造型表存入盘中，名为M
编制绘图程序并存盘	]	10 POKE 232,0: POKE 233,3	将图形表首地址$0300置入内存232,233单元
		20 HGR2 30 HCOLOR=3	高分辨率绘图 图形颜色为白色
		40 FOR I=1 TO 40 50 ROT=I	将图形旋转，角度=$(I\times\frac{90}{16})$
		60 SCALE=I	将图放大I倍
		70 DRAW 1 AT 140,80	绘制第一个造型，中心坐标在(140，80)
		80 NEXT I 90 END	
		SAVE N	绘图程序存盘，名为N
绘图并打印输出		BLOAD M↙	将造型表装入内存
		LOAD N↙	将绘图程序装入内存
		RUN↙	进行绘图
		PR#1↙	启动打印机
		POKE 1913，2↙ PRINT CHR$(17)↙	将绘图结果打印输出

```
90  READ X: POKE I,X
100  NEXT I
110  POKE 232,0: POKE 233,3
120  HOME : HGR2 : HCOLOR= 3: SCALE= 3: ROT=
 45
130  FOR X = 10 TO 200 STEP 10
140  FOR I = 1 TO 5
150  DRAW I AT X,100
160  XDRAW I AT X,100
170  NEXT I
180  NEXT X
190  GOTO 130
```

以上介绍的造型生成程序,数据都是用10进制数存贮的。

下面介绍一种用二进制造型表的方法,绘制高分辨图形,造型只有一个，是一个正方形，但利用程序控制，使这个正方形显示出方块旋转螺线图。整个程序编制的详细说明，见表9.2。

(5) 造型表应用

下面给出一个ASCII码造型表程序H93， 程序中共有96个造型，运行车程序后，稍等一段时间，即在屏幕上显示所有96个造型图形，若采用下述命令,即可打印输出如图9.7显示的反相显示图形。

PR#1↙

POKE 1913,2↙

POKE 1913，34↙

PRINT CHR$ (17)↙

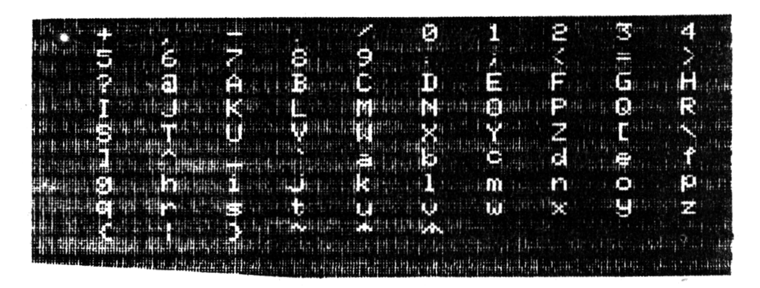


图 9.7

```
]LIST

5  REM H93
10  REM  ASCII SHAPES
20  REM NUMBERS SHAPE
30  GOSUB 280
40  HGR2 : HCOLOR= 3
50  ROT= 0: SCALE= 1
60  FOR I = 1 TO N
70  IF I > 90 THEN 250
80  IF I > 80 THEN 240
90  IF I > 70 THEN 230
100  IF I > 60 THEN 220
110  IF I > 50 THEN 210
120  IF I > 40 THEN 200
130  IF I > 30 THEN 190
140  IF I > 20 THEN 180
150  IF I > 10 THEN 170
160  DRAW I AT 22 * I,10: GOTO 260
170  DRAW I AT 22 * (I - 10),20: GOTO 260
180  DRAW I AT 22 * (I - 20),30: GOTO 260
```

```
190  DRAW I AT 22 * (I - 30),40: GOTO 260
200  DRAW I AT 22 * (I - 40),50: GOTO 260
210  DRAW I AT 22 * (I - 50),60: GOTO 260
220  DRAW I AT 22 * (I - 60),70: GOTO 260
230  DRAW I AT 22 * (I - 70),80: GOTO 260
240  DRAW I AT 22 * (I - 80),90: GOTO 260
250  DRAW I AT 22 * (I - 90),100: GOTO 260
260  NEXT I
270  END
280  REM MULTI SHAPE
290 S = 24576: REM  FIRST ADDRESS OF SHAPE T
ABLE
300 S1 =  INT (S / 256):S2 = S - S1 * 256
310  POKE 232,S2: POKE 233,S1
320  READ N:K = N: REM SHAPE DEFINIT NUMBERS
330  POKE S,N: POKE S + 1,0
340 M = S + 2 * (N + 1):S2 = S + 2
350 D = M - S
360  IF D > 255 THEN 390
370  POKE S2,D: POKE S2 + 1,0
380  GOTO 410
390 D1 =  INT (D / 256):D2 = D - D1 * 256
400  POKE S2,D2: POKE S2 + 1,D1
410 S2 = S2 + 2
420  READ A
430  IF A = 8 THEN 500
440  READ B
450  IF B = 8 THEN 490
460 X = B * 8 + A
470  POKE M,X:M = M + 1
480  GOTO 420
```

```
490  POKE M,A:M = M + 1
500  POKE M,0:M = M + 1
510 K = K - 1
520  IF K = 0 THEN  RETURN
530  GOTO 350
540  DATA 96
550  DATA 1,1,6,6,6,6,2,2,6,8: REM "!"
560  DATA  1,6,6,5,1,4,4,4,8: REM """
570  DATA  1,5,1,6,7,3,7,2,5,5,5,5,6,3,7,3,7
,2,5,5,5,5,6,3,7,3,6,5,1,5,8: REM "#"
580  DATA 1,5,1,2,7,7,7,7,2,5,1,5,2,7,7,6,1,
5,1,6,3,7,7,7,6,1,1,6,8: REM "$"
590  DATA 5,6,7,5,1,1,1,6,3,6,3,6,3,7,2,5,1,
1,5,6,7,7,8: REM "%"
600  DATA 1,5,2,7,3,6,5,1,6,3,7,2,5,1,5,1,6,
3,7,3,3,6,1,5,5,1,5,8:"&"
610  DATA 1,1,1,6,6,6,8:"'"
620  DATA 1,1,1,1,6,3,6,3,6,6,5,2,5,2,5,8:"(
"

630  DATA 1,6,1,6,1,6,6,6,3,6,3,6,8:")"
640  DATA 1,1,5,1,2,7,3,7,3,6,1,5,5,5,2,7,7,
7,7,6,1,5,5,5,2,7,3,7,3,6,1,1,6,8:"*"
650  DATA 2,2,2,5,5,4,4,5,1,2,2,7,6,3,6,6,8:
"+"
660  DATA 2,2,2,2,2,1,6,6,3,6,8:","
670  DATA 2,2,2,5,5,5,5,5,8:"-"
680  DATA 2,2,2,2,2,2,1,6,8:"."
690  DATA 1,1,1,1,6,3,6,3,6,3,6,3,6,8:"/"
700  DATA 1,5,5,5,2,6,6,6,6,6,3,7,7,7,0,4,4,
4,4,5,1,1,2,6,3,6,3,6,8:"0"
710  DATA 1,1,6,7,6,1,6,6,6,6,7,5,5,5,8: REM
```

```
 "1"
720  DATA 2,5,0,5,5,5,2,6,6,3,7,7,7,2,6,6,5,
5,5,5,5,8: REM "2"
730  DATA 5,5,5,5,6,3,7,2,5,2,5,2,6,6,3,7,7,
7,0,4,8  : REM "3"
740  DATA 2,2,2,6,5,5,6,2,1,4,4,5,4,3,4,4,4,
7,2,7,2,7,8: REM "4"
750  DATA 5,5,5,5,6,3,3,3,3,6,5,5,5,5,2,6,6,
6,3,7,7,7,0,5,8: REM "5"
760  DATA 1,1,5,6,3,3,6,3,6,5,5,5,5,2,6,6,3,
7,7,7,0,4,4,8: REM "6"
770  DATA 5,5,5,5,6,6,3,6,3,6,3,6,3,6,5,8: R
EM "7"
780  DATA 1,5,5,5,2,6,6,2,6,6,3,7,7,7,0,4,4,
1,5,5,4,3,3,3,4,4,8: REM "8"
790  DATA 1,5,5,5,2,6,7,3,3,3,4,5,2,2,5,5,5,
6,6,3,6,3,7,7,8: REM "9"
800  DATA 2,2,2,1,6,2,6,8:":"
810  DATA 2,2,1,1,6,2,6,6,3,6,8:";"
820  DATA 1,1,1,6,3,6,3,6,3,6,1,5,2,5,2,5,8:
"<"
830  DATA 2,2,5,5,5,5,6,2,7,7,7,7,7,8:"="
840  DATA 1,6,1,6,1,6,1,6,3,6,3,6,3,6,8:">"
850  DATA 2,5,0,5,5,5,2,6,6,3,6,3,6,2,6,8:"?
"
860  DATA 2,5,0,5,5,5,6,6,6,6,6,6,7,7,7,4,3,
4,4,1,5,6,6,6,8:"@"
870  DATA 1,1,5,2,7,3,7,2,6,6,6,6,5,1,1,1,4,
4,4,4,7,2,2,7,7,7,8:"A"
880  DATA 5,5,5,5,2,7,3,3,6,5,1,1,6,3,7,7,6,
6,7,2,5,5,5,5,0,4,4,8:"B"
```

```
890  DATA 1,5,5,5,2,6,2,2,2,6,3,7,7,7,0,4,4,
4,4,4,8:"C"
900  DATA 5,5,5,5,2,6,6,6,6,6,3,7,7,7,4,1,4,
4,4,4,4,8:"D"
910  DATA 5,5,5,5,6,3,3,3,3,6,6,5,5,5,6,3,3,
3,6,6,5,5,5,5,5,8:"E"
920  DATA 5,5,5,5,6,3,3,3,3,6,6,5,5,5,6,3,3,
3,6,6,5,8:"F"
930  DATA 1,5,5,5,6,2,2,6,6,6,7,7,7,4,3,4,4,
4,4,5,1,1,2,2,5,5,8:"G"
940  DATA 6,6,6,6,6,6,5,1,1,1,4,4,4,4,4,4,7,
2,2,2,7,7,7,8:"H"
950  DATA 1,5,5,6,3,6,6,6,6,7,2,5,5,5,8:"I"
960  DATA 1,1,1,1,6,6,6,6,6,6,3,7,7,7,0,4,8:
"J"
970  DATA 6,6,6,6,6,6,5,1,1,1,4,3,4,3,4,3,4,
1,4,1,4,1,4,,8:"K"
980  DATA 6,6,6,6,6,6,5,5,5,5,5,8:"L"
990  DATA 6,6,6,6,6,6,5,1,1,1,4,4,4,4,4,4,7,
2,7,3,6,1,5,8:"M"
1000  DATA 6,6,6,6,6,6,5,1,1,1,4,4,4,4,4,4,7
,3,3,2,6,1,6,1,6,8:"N"
1010  DATA 1,5,5,5,2,6,6,6,6,6,3,7,7,7,0,4,4
,4,4,4,8:"O"
1020  DATA 5,5,5,5,2,6,6,3,7,7,7,4,4,5,2,2,2
,3,6,6,6,8:"P"
1030  DATA 1,5,5,5,2,6,6,7,3,3,3,4,4,5,2,2,2
,3,5,1,5,1,6,3,7,3,3,6,1,5,5,1,5,8:"Q"
1040  DATA 5,5,5,5,2,6,6,3,7,7,7,4,4,5,2,2,2
,3,6,6,5,1,1,1,4,3,4,3,4,8:"R"
1050  DATA 1,5,5,5,2,7,3,3,3,6,6,1,5,5,5,2,6
```

```
,7,3,3,3,6,1,5,5,5,8:"S"
1060  DATA 5,5,5,5,6,3,3,6,6,6,6,6,6,8:"T"
1070  DATA 6,6,6,6,6,5,1,1,1,4,4,4,4,4,7,2,2
,2,2,2,2,7,7,7,8:"U"
1080  DATA 6,6,5,1,1,1,4,4,7,2,2,2,7,3,6,5,1
,6,3,6,6,8:"V"
1090  DATA 6,6,6,6,5,1,5,1,6,7,7,3,7,6,5,1,1
,1,4,4,4,4,4,4,4,8:"W"
1100  DATA 6,5,1,1,1,4,7,2,2,7,3,6,1,5,2,7,3
,7,2,6,5,1,1,1,4,4,8:"X"
1110  DATA 6,5,1,1,1,4,7,2,2,7,3,6,1,6,6,6,6
,8:"Y"
1120  DATA 5,5,5,5,6,7,2,7,2,7,2,7,2,6,5,5,5
,5,5,8:"Z"
1130  DATA 1,5,5,6,3,3,6,6,6,6,6,5,5,5,8:"["
1140  DATA 2,5,2,5,2,5,2,5,2,5,8:"\"
1150  DATA 1,5,5,6,6,6,6,6,6,7,7,7,8:"]"
1160  DATA 1,1,5,2,7,3,7,2,5,1,1,1,5,8:"^"
1170  DATA 2,2,2,2,2,2,5,5,5,5,5,8:"_"
1180  DATA 1,1,5,2,5,8:"`"
1190  DATA 2,2,1,5,5,5,2,6,7,7,7,7,2,6,1,5,5
,5,4,4,8:"a"
1200  DATA 6,6,5,1,5,5,2,7,3,3,7,6,5,1,1,1,6
,7,3,3,7,6,5,1,5,5,8:"b"
1210  DATA 2,1,5,5,5,2,7,3,3,3,6,6,5,1,1,1,6
,3,7,7,7,8:"c"
1220  DATA 1,1,1,1,6,6,7,3,7,7,2,5,1,1,5,6,7
,3,3,3,6,5,1,1,5,6,7,3,7,7,8:"d"
1230  DATA 2,2,1,5,5,5,2,7,3,3,3,6,5,5,5,5,6
,3,3,3,3,6,1,5,5,5,8:"e"
1240  DATA 1,1,1,5,2,7,3,7,2,5,5,6,3,6,6,6,6
```

```
,8:"f"
1250  DATA 2,1,5,5,1,6,7,7,3,3,6,5,1,1,5,6,7
,3,7,7,2,1,1,1,1,6,7,3,3,3,6,1,5,5,5,8:"g"
1260  DATA 6,6,5,1,5,5,2,6,6,6,7,3,3,3,4,4,4
,5,5,8:"h"
1270  DATA 1,1,6,3,2,5,6,6,6,7,2,5,5,5,8:"i"
1280  DATA 1,1,1,6,2,,6,6,6,7,3,3,3,6,1,5,5,
5,8:"j"
1290  DATA 6,6,5,1,1,6,3,7,3,6,5,5,2,7,3,6,5
,1,1,5,8:"k"
1300  DATA 1,5,6,6,6,6,6,7,2,5,5,5,8:"l"
1310  DATA 2,2,5,5,1,5,2,7,3,7,3,6,5,1,5,1,6
,7,3,7,3,6,5,1,5,1,5,8:"m"
1320  DATA 2,2,5,1,5,5,2,7,3,3,7,6,6,6,5,1,1
,1,4,4,4,4,8:"n"
1330  DATA 2,2,1,5,5,5,2,6,6,6,3,7,7,7,0,4,4
,4,8:"o"
1340  DATA 2,5,1,5,5,2,7,3,3,7,6,5,1,1,1,6,7
,3,3,7,6,5,1,5,5,2,,3,3,3,3,6,6,6,8:"p"
1350  DATA 2,1,5,5,1,6,7,7,3,3,6,5,1,1,1,6,7
,7,3,3,6,1,5,5,1,6,6,6,8:"q"
1360  DATA 2,2,5,1,5,5,2,7,3,3,7,6,6,6,6,8:"
r"
1370  DATA 2,2,1,5,5,5,6,3,3,3,3,6,1,5,5,5,2
,6,3,7,7,7,7,8:"s"
1380  DATA 1,6,5,1,2,7,7,7,6,1,6,6,6,1,4,1,5
,8:"t"
1390  DATA 2,2,6,6,6,6,1,5,5,1,4,7,4,1,4,4,4
,8:"u"
1400  DATA 2,2,6,6,6,1,6,1,4,1,4,1,4,4,4,,8:
"v"
```

```
1410  DATA 2,2,6,6,6,6,1,5,1,5,0,7,3,4,5,1,4
,4,4,8:"w"
1420  DATA 2,2,5,1,1,1,6,3,7,3,6,1,5,2,7,3,7
,2,5,1,1,1,5,8:"x"
1430  DATA 2,6,6,6,2,2,6,1,5,5,4,1,4,4,4,4,4
,7,2,2,2,7,7,7,8:"y"
1440  DATA 2,2,5,5,5,5,6,3,6,3,6,3,7,2,5,5,5
,5,5,8:"z"
1450  DATA 1,1,1,5,6,3,3,6,7,2,6,1,6,6,1,5,5
,8:"["
1460  DATA 1,1,6,6,6,2,6,6,6,8:" "
1470  DATA 1,5,5,2,6,6,1,6,3,6,6,3,7,7,8:"]"
1480  DATA 2,4,1,6,1,5,1,6,3,7,8:" "
1490  DATA 2,2,5,1,5,1,4,3,7,7,4,1,6,8:" "
1500  DATA 1,1,1,5,2,7,3,7,2,5,1,5,1,5,2,7,3
,3,7,3,3,7,8
```

整个程序共分三大段:

① 10—270句:是主程序，运行后光调用各个造型图形，然后利用绘图语句在屏幕上显示所有造形。

② 280—530句:是造型表生成子程序，造型表首地址存放在第二页高分辨率图形区的上部24576地方,对应的十六进制数为＄6000。

③ 540—1500句：是存贮造型图形的数据区。540句DATA中的96是造型个数。550句以后每一个DATA语句存贮一个造型数据（以八进制数表示),并以造型结束符8作为结尾。

注：利用本造型表，可以在高分辨图形区的任意位置加

注英文说明。关于它的使用实例见计算机辅助教学。

3. 计算机辅助教学（绘电路图）

这里介绍一个绘制电路图的实用程序H94，它可作为电子线路教学的辅助演示，是计算机辅助教学的一种尝试。

利用中华学习机高分辨率作图功能，可以绘制各种各样的模拟电路和数字电路原理图或逻辑符号。利用造型表技术，在电路图形的适当地方加注元件参数、文字符号、计算公式、布尔表达式等。从而在电子线路的电化教学中，做到图文并茂，丰富多采。

程序H94共画了与门、或非门、或门、与或非门、异或门、R-S触发器，共发射放大电路等8个电路。实际上，可以以本程序为模式，绘制出各种各样的电子线路。此外，还可以绘制程序设计框图，印刷线路板等。

为了在电路图形的任意位置加注说明性符号，必须利用前节介绍的造型表（ASCII SHAPES），为此，应安排以下两个语句：

POKE 232, 0

POKE 233, 96

这是因为232和233两个单元是专门用来存放造型表首地址的。

从前节程序中知道，ASCII码造型首地址是十进制数24576，对应的十六进制数为＄6000。POKE 232后面应放造型表首地址的低字节（十进制数），所以是0；POKE 233后面应放造型表首地址的高字节（十进制数）96，96是十六进制数60的十进制数表示。

只有这样，将图形表送入内存，并将其首地址记入232和233单元中，上节造型表才能使用。

另外，为了在电路图的适当位置加注说明性符号，本程序中安排了诸如X $ ="P":XZ=110: YZ=0:GOSUB 1810等语句。这个意思是在坐标（110，0）的位置上，标上字符"P"。1810以后的语句，是计算机自动给出X $ 中的字符。

H94在120—150句中安排了总控，而在1700—1800句安排了是否要打印的控制，如要打印按"Y"键，不需要打印按空格键。

程序输出打印既有图形，又有英文注释；既有布不表达式，又有中文汉字的说明。

程序H94运行方法是：

· 首先运行造型表，RUN ASCII SHAPES↙ 稍等片刻，出现ASCII 字符表，按CTRL-RESET中断。

· 调进计算机辅助教学（绘电路图）程序，并运行之。

· 出现？后，键入1—8中任一个数字，计算机自动绘图，键码代号如下：

计算机辅助教学(绘电路图)

1.【与门电路—全高出高，有低出低】

2.【或非门电路—有高出低，全低出高】

3.【或门电路—有高出高，全低出低】

4.【与非门电路—全高出低，有低出高】

5.【与或非门电路—先与后或再非】

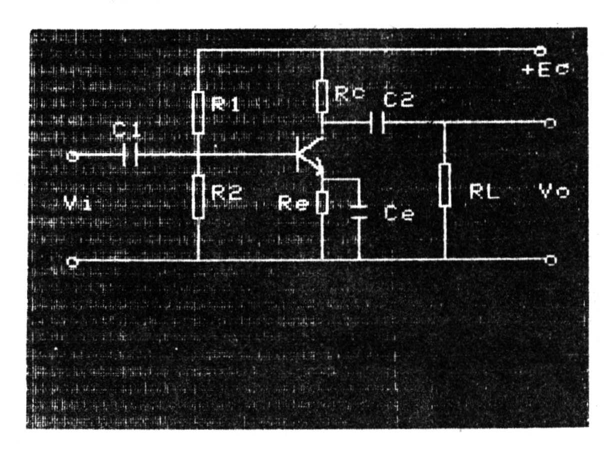
6.【异或门电路—不同出高，相同出低】

7.【RS—触发器】

8.【电流负反馈电路】

·出现“DO-YOU-WANT-PRINT(Y/N)”后，若要打印，按Y键，反之按N键或空格键。

若将程序中POKE1913，2改为POKE1913，34，则打印反相图形，如图9.8。



【电流负反馈电路】

图 9.8

·按9↙，停机。

计算机辅助教学（绘电路图）清单（见H94）及运行结果（见图9.9—9.16）：

```
]LIST

2  REM H94
5  GOSUB 1900
10  REM 计算机辅助教学(绘电路图)
20  DIM A$(10)
30  FOR I = 1 TO 8
40  READ A$(I)
50  NEXT I
60  POKE 232,0: POKE 233,96
70  DATA 【与门电路—全高出高，有低出低】,【
或非门电路—有高出低，全低出高】
80  DATA 【或门电路—有高出高，全低出低】,【
与非门电路—全高出低，有低出高】
90  DATA  【与或非门电路—先与后或再非】,【
异或门电路—不同出高，相同出低】
100  DATA 【RS—触发器】,【电流负反馈电路】
110  POKE 232,0: POKE 233,96
120  INPUT X
130  IF X = 9 THEN  END
140  ON X GOSUB 160,280,430,570,710,920,1030
,1260,900
150  GOTO 70
160  HGR2 : HCOLOR= 3
170  HPLOT 110,8 TO 110,50
180  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
190  HPLOT 80,100 TO 80,150
200  HPLOT 140,100 TO 140,150
210 X$ = "P":XZ = 110:YZ = 0: GOSUB 1810
220 X$ = "A":XZ = 80:YZ = 155: GOSUB 1810
```

```
230 X$ = "B":XZ = 140:YZ = 155: GOSUB 1810
240 X$ = "AND-CIRCUIT":XZ = 75:YZ = 75: GOSU
B 1810
250 X$ = "P=A.B":XZ = 20:YZ = 20: GOSUB 1810
260  GOSUB 1700
270  RETURN
280  HGR2 : HCOLOR= 3
290  HPLOT 110,8 TO 110,44

]LIST300,

300 X$ = "O":XZ = 111:YZ = 43: GOSUB 1810
310  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
320  HPLOT 80,100 TO 80,150
330  HPLOT 140,100 TO 140,150
340 X$ = "+":XZ = 110:YZ = 75: GOSUB 1810
350 X$ = "P":XZ = 110:YZ = 0: GOSUB 1810
360 X$ = "A":XZ = 80:YZ = 155: GOSUB 1810
370 X$ = "B":XZ = 140:YZ = 155: GOSUB 1810
380 X$ = "P=A+B":XZ = 20:YZ = 20: GOSUB 1810
390  HPLOT 30,18 TO 50,18
400 X$ = "NOR-CIRCUIT":XZ = 73:YZ = 85: GOSU
B 1810
410  GOSUB 1700
420  RETURN
430  HGR2 : HCOLOR= 3
440  HPLOT 110,8 TO 110,50
450  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
460  HPLOT 80,100 TO 80,150
470  HPLOT 140,100 TO 140,150
```

```
480  HPLOT 106,75 TO 114,75
490  HPLOT 110,70 TO 110,80
500 X$ = "P":XZ = 110:YZ = 0: GOSUB 1810
510 X$ = "A":XZ = 80:YZ = 155: GOSUB 1810
520 X$ = "B":XZ = 140:YZ = 155: GOSUB 1810
530 X$ = "OR-CIRCUIT":XZ = 75:YZ = 85: GOSUB
 1810
540 X$ = "P=A+B":XZ = 20:YZ = 20: GOSUB 1810
550  GOSUB 1700
560  RETURN
570  HGR2 : HCOLOR= 3
580  HPLOT 110,8 TO 110,40
590  HPLOT 108,40 TO 112,40 TO 112,50 TO 108
,50 TO 108,40
600  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
610  HPLOT 80,100 TO 80,150
620  HPLOT 140,100 TO 140,150
630 X$ = "P":XZ = 110:YZ = 0: GOSUB 1810
640 X$ = "A":XZ = 80:YZ = 155: GOSUB 1810
650 X$ = "B":XZ = 140:YZ = 155: GOSUB 1810
660 X$ = "NAND-CIRCUIT":XZ = 74:YZ = 75: GOS
UB 1810
670 X$ = "P=A.B":XZ = 20:YZ = 20: GOSUB 1810
680  HPLOT 30,18 TO 50,18
690  GOSUB 1700
700  RETURN
710  HGR2 : HCOLOR= 3
720  HPLOT 110,8 TO 110,40
730  HPLOT 108,40 TO 112,40 TO 112,50 TO 108
,50 TO 108,40
```

```
740  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
750  HPLOT 50,100 TO 170,100 TO 170,150 TO 5
0,150 TO 50,100
760  HPLOT 110,100 TO 110,150
770  HPLOT 70,150 TO 70,170
780  HPLOT 90,150 TO 90,170
790  HPLOT 130,150 TO 130,170
800  HPLOT 150,150 TO 150,170
810  HPLOT 106,75 TO 114,75
820  HPLOT 110,70 TO 110,80
830 X$ = "P":XZ = 110:YZ = 0: GOSUB 1810
840 X$ = "A":XZ = 70:YZ = 175: GOSUB 1810
850 X$ = "B":XZ = 90:YZ = 175: GOSUB 1810
860 X$ = "C":XZ = 130:YZ = 175: GOSUB 1810
870 X$ = "D":XZ = 150:YZ = 175: GOSUB 1810
880 X$ = "ANOR-CIRCUIT":XZ = 70:YZ = 85: GOS
UB 1810
890 X$ = "P=AB+CD":XZ = 20:YZ = 20: GOSUB 18
10
900  HPLOT 30,18 TO 63,18: GOSUB 1700
910  RETURN
920  HGR2 : HCOLOR= 3
930  HPLOT 110,8 TO 110,50
940  HPLOT 50,50 TO 170,50 TO 170,100 TO 50,
100 TO 50,50
950  HPLOT 80,100 TO 80,150
960  HPLOT 140,100 TO 140,150
970  HPLOT 106,75 TO 114,75
980  HPLOT 110,70 TO 110,80
990  HPLOT 104,65 TO 116,65 TO 116,85 TO 104
```

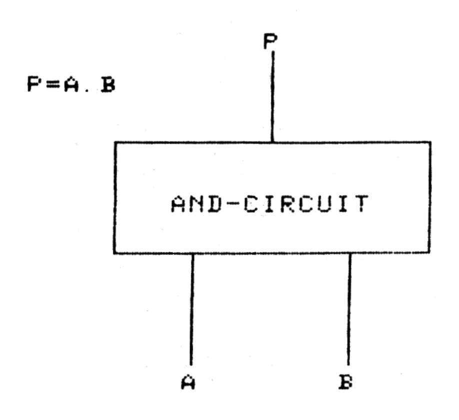
```
,85 TO 104,65
1000 X$ = "EXOR-CIRCUIT":XZ = 73:YZ = 88: GO
SUB 1810
1010  GOSUB 1700
1020  RETURN
1030  HGR2 : HCOLOR= 3
1040  HPLOT 70,20 TO 70,50
1050  HPLOT 150,20 TO 150,50
1060  HPLOT 50,50 TO 170,50 TO 170,100 TO 50
,100 TO 50,50
1070  HPLOT 70,100 TO 70,140
1080  HPLOT 150,100 TO 150,140
1090  HPLOT 110,100 TO 110,167
1100  HPLOT 104,100 TO 110,95 TO 116,100
1110  HPLOT 68,145 TO 72,145
1120  HPLOT 68,145 TO 68,149 TO 72,149 TO 72
,153 TO 68,153
1130  HPLOT 148,145 TO 152,145 TO 152,149 TO
 148,149 TO 152,153
1140  HPLOT 148,145 TO 148,153
1150  HPLOT 104,170 TO 108,170
1160  HPLOT 104,170 TO 104,178 TO 108,178
1170  HPLOT 110,170 TO 110,178
1180  HPLOT 110,170 TO 114,170 TO 114,174 TO
 110,174
1190  HPLOT 68,60 TO 72,60 TO 72,68 TO 68,68
 TO 68,60
1200  HPLOT 70,64 TO 74,68
1210  HPLOT 148,60 TO 152,60 TO 152,68 TO 14
8,68 TO 148,60
1220  HPLOT 148,57 TO 152,57: HPLOT 150,64 T
```

```
O 154,68
1230 X$ = "R-S-FLIP-FLOPS":XZ = 68:YZ = 85:
GOSUB 1810
1240  GOSUB 1710
1250  RETURN
1260  HGR2 : HCOLOR= 3
1270  HPLOT 70,10 TO 70,30
1280  HPLOT 68,30 TO 72,30 TO 72,50 TO 68,50
 TO 68,30
1290  HPLOT 70,50 TO 70,70
1300  HPLOT 68,70 TO 72,70 TO 72,90 TO 68,90
 TO 68,70
1310  HPLOT 70,90 TO 70,110
1320  HPLOT 20,60 TO 40,60: HPLOT 45,60 TO 1
10,60
1330  HPLOT 40,55 TO 40,65: HPLOT 45,55 TO 4
5,65
1340  HPLOT 110,52 TO 110,68: HPLOT 110,60 T
O 120,52: HPLOT 110,60 TO 120,68
1350  HPLOT 120,68 TO 118,62: HPLOT 120,68 T
O 118,70
1360  HPLOT 120,68 TO 120,78: HPLOT 118,78 T
O 122,78 TO 122,88 TO 118,88 TO 118,78
1370  HPLOT 120,89 TO 120,110
1380  HPLOT 120,72 TO 135,72 TO 135,85
1390  HPLOT 132,85 TO 138,85: HPLOT 132,90 T
O 138,90
1400  HPLOT 135,91 TO 135,110
1410  HPLOT 20,110 TO 210,110
1420  HPLOT 70,10 TO 210,10
1430  HPLOT 120,50 TO 120,40
```

```
1440  HPLOT 118,40 TO 122,40 TO 122,25 TO 11
8,25 TO 118,40
1450  HPLOT 120,25 TO 120,10
1460  HPLOT 120,45 TO 140,45: HPLOT 140,40 T
O 140,50
1470  HPLOT 145,40 TO 145,50
1480  HPLOT 145,45 TO 170,45
1490  HPLOT 170,45 TO 170,65
1500  HPLOT 168,65 TO 172,65 TO 172,85 TO 16
8,85 TO 168,65
1510  HPLOT 170,85 TO 170,110
1520 X$ = "o":XZ = 20:YZ = 57: GOSUB 1810
1530 X$ = "o":XZ = 20:YZ = 107: GOSUB 1810
1540 X$ = "o":XZ = 210:YZ = 7: GOSUB 1810
1550 X$ = "V1":XZ = 20:YZ = 80: GOSUB 1810
1560 X$ = "C1":XZ = 40:YZ = 45: GOSUB 1810
1570 X$ = "R1":XZ = 80:YZ = 33: GOSUB 1810
1580 X$ = "R2":XZ = 80:YZ = 75: GOSUB 1810
1590 X$ = "Rc":XZ = 130:YZ = 28: GOSUB 1810
1600 X$ = "C2":XZ = 150:YZ = 31: GOSUB 1810
1610 X$ = "Re":XZ = 107:YZ = 80: GOSUB 1810
1620 X$ = "Ce":XZ = 150:YZ = 85: GOSUB 1810
1630 X$ = "RL":XZ = 185:YZ = 75: GOSUB 1810
1640 X$ = "+Ec":XZ = 213:YZ = 17: GOSUB 1810
:X$ = "Vo":XZ = 213:YZ = 75: GOSUB 1810
1650  HPLOT 170,45 TO 210,45
1660 X$ = "o":XZ = 214:YZ = 42: GOSUB 1810
1670 X$ = "o":XZ = 214:YZ = 107: GOSUB 1810
1680  GOSUB 1710
1690  RETURN
1700 X$ = "DO-YOU-WANT-PRINT(Y/N)":XZ = 120:
```

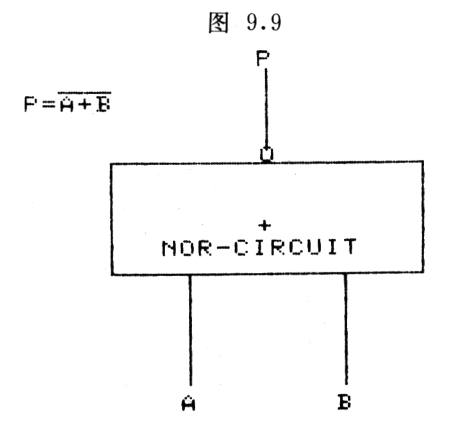
```
YZ = 35: GOSUB 1810
1710  GOTO 1730
1720 X$ = "DO-YOU-WANT-TO-PRINT(Y/N)":XZ = 7
0:YZ = 20: GOSUB 1810
1730  GET A$: IF A$ = "N" OR A$ = " " THEN
TEXT : HOME : PRINT "GOOD BYE!": RETURN
1740  IF A$ < > "Y" THEN  PRINT  CHR$ (7):
RETURN
1750  IF X = 8 THEN 1770
1760  HCOLOR= 0:X$ = "DO-YOU-WANT-PRINT(Y/N)
":XZ = 120:YZ = 35: GOSUB 1810
1770  PR# 1: POKE 1913,2
1780  POKE 1913,2: PRINT  CHR$ (17): PR# 0
1790  PR# 3: PRINT : POKE 1659,5: PRINT  TAB
( 8);A$(X): POKE 1659,0: TEXT
1800  RETURN
1810  ROT= 0:XX =  LEN (X$): IF XX = 3 THEN
XZ = XZ - 11: GOTO 1840
1820  IF XX = 2 THEN XZ = XZ - 4: GOTO 1840
1830 XZ = XZ - 3
1840  FOR I1 = 0 TO XX - 1:AS =  ASC ( MID$
(X$,I1 + 1,1)) - 32: DRAW AS AT XZ + I1 * 7,
YZ: NEXT I1: RETURN
1900  PRINT : PRINT : PRINT
1950  PRINT "计算机辅助教学(绘电路图)"
1970  PRINT : PRINT
2000  PRINT "1.【与门电路—全高出高，有低出
低】"
2050  PRINT : PRINT
2100  PRINT "2.【或非门电路—有高出低，全低
出高】"
```

```
2150  PRINT : PRINT
2200  PRINT "3.【或门电路一有高出高，全低出低】"
2250  PRINT : PRINT
2300  PRINT "4.【与非门电路一全高出低，有低出高】"
2350  PRINT : PRINT
2400  PRINT "5.【与或非门电路一先与后或再非】"
2450  PRINT : PRINT
2500  PRINT "6.【异或门电路一不同出高，相同出低】"
2550  PRINT : PRINT
2600  PRINT "7.【RS一触发器】"
2650  PRINT : PRINT
2700  PRINT "8.【电流负反馈电路】"
2750  PRINT : PRINT
2800  RETURN
```



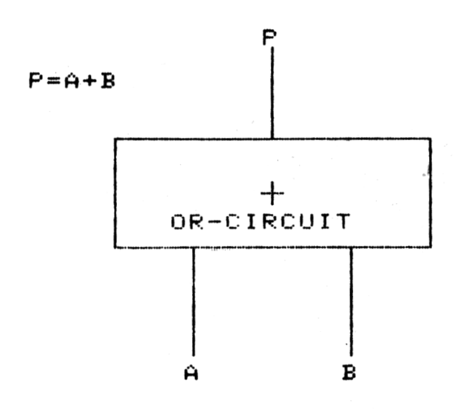
【与门电路—全高出高，有低出低】

图 9.9



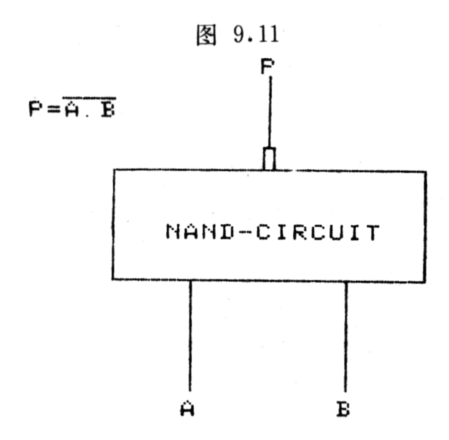
【或非门电路—有高出低，全低出高】

图 9.10



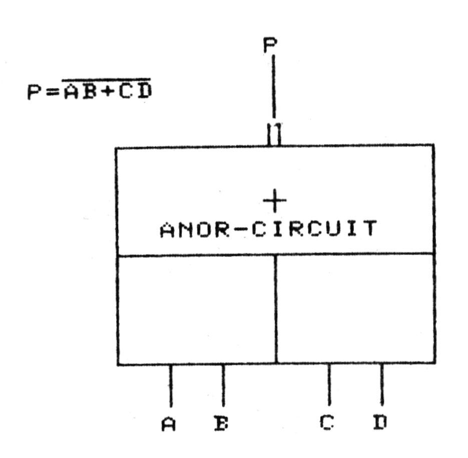
【或门电路—有高出高，全低出低】

图 9.11



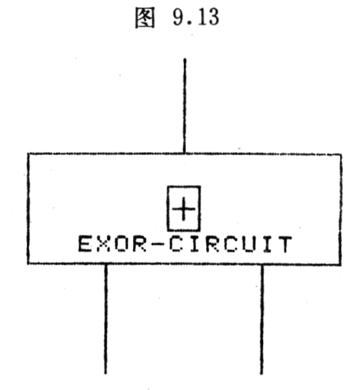
【与非门电路—全高出低，有低出高】

图 9.12



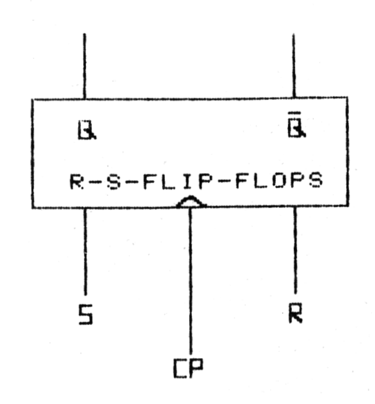
【与或非门电路—先与后或再非】

图 9.13



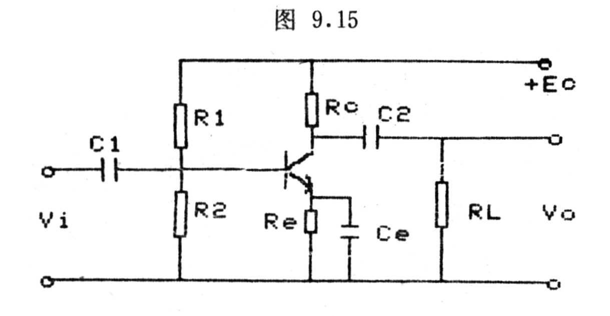
【异或门电路—不同出高，相同出低】

图 9.14



【RS—触发器】

图 9.15



【电流负反馈电路】

图 9.16

4. 全屏幕绘图软件

下面介绍的是一个实用的全屏幕绘图软件，其特点是：

· 用有关按键控制光点移动方向，进行绘图或写字。图

象水平的高低，由操作者的绘图技术决定。

·使用简单，操作者可以根据中文菜单提示，进行各种操作，只要稍事熟悉即可运用自如。

·删改灵活，画的不满意的地方，可以随意擦涂更改。

·存贮方便，完成或尚未完成的作品，可以随时存贮，使用时可以方便地调出，或继续完成，或打印输出。

·打印多样，需要打印的图案，有四种方式选择：正常、放大、反相、反相放大。

·既可以随时擦涂全部图象，也可以在屏幕上保存图象。

使用方法：

·运行本软件后，即出现以下中文菜单，菜单上详细注明各种按键的作用，如按Y键，光点向左上方移动并作图。擦涂先按0键再按其它键，恢复画图先按L键，再按其它键。

全屏幕绘图软件

Y—左上	G—左下
B—右下	H—右上
I—向上	M—向下
K—向右	J—向左
L—画图	O—擦图
C—清除	T—保留
S—存盘	A—调盘
W—打印	E—结束

按空格键运行

·菜单出现后，按空格(SPACE)键，即出现一个沿屏幕边沿的方框，光点在屏幕正中央闪烁，表明进入绘图

状态。

·画好图象后，按S 键，屏幕提出“请输入文件名”,此时输入文件名（例如C -F ）并回车后，机器一面自动存盘，一面给出“正在存盘，请稍等……”的信息，存盘完成后，机器提示“存盘结束”,可以进行其它功能操作。

·调出图象只要按A 键，机器立即提示“请输入调出的文件名:”，此时输入文件名并回车，机器一面调盘，一面给出“正在调盘，请稍等……”的信息，调盘完成后，机器也会打出“调盘结束”的信息，再进行其它操作。

·在绘图状态下，任何时刻按C 键即可清除画面，并重新进入绘图状态，原图象丢失；按T 键，清除画面，重新进入绘图状态后，原图象保持；按E 键结束停机。

·打印图象按W键，有四种方式，按 1 是正常打印，按2是放大打印，按3是反相打印，按 4 是反相放大打印。每种打印方式都是先按W键再按数字键。图象打印完成后，机器给出“硬拷贝完成”的提示。

下面给出完整的程序清单(见H 95)及运行的图象实例。

《本软件适应机种：中华学习机、APPLE 机、紫金II》

```
]LIST

5  REM H95
10  REM    全屏幕绘图软件
20  HGR :X = 140:Y = 95:B$ = "L"
30  PR# 3: PRINT : HOME
40  GOTO 180
50  PRINT  CHR$ (18)
60  VTAB 1: HTAB 10: PRINT "全屏幕绘图软件"
```

```
70  HTAB 2: PRINT "Y—左上";: HTAB 25: PRINT
 "G—左下"
80  HTAB 2: PRINT "B—右下";: HTAB 25: PRINT
 "H—右上"
90  HTAB 2: PRINT "I—向上";: HTAB 25: PRINT
 "M—向下"
100  HTAB 2: PRINT "K—向右";: HTAB 25: PRIN
T "J—向左"
110  HTAB 2: PRINT "L—画图";: HTAB 25: PRIN
T "O—擦图"
120  HTAB 2: PRINT "C—清除";: HTAB 25: PRIN
T "T—保留"
130  HTAB 2: PRINT "S—存盘";: HTAB 25: PRIN
T "A—调盘"
140  HTAB 2: PRINT "W—打印";: HTAB 25: PRIN
T "E—结束"
150  VTAB 10: HTAB 10: PRINT "按空格键运行";
160  GET E$: IF E$ = " " THEN  RETURN
170  GOTO 160
180  GOSUB 50
190  GOTO 390
200 W1 =  PEEK ( - 16384): IF W1 > 127 THEN
A$ =  CHR$ (W1 - 128): POKE  - 16368,0: GOTO
 240
210  HPLOT X,Y: HCOLOR= 0: HPLOT X,Y: HCOLOR
= 3
220  IF B$ = "L" THEN  HPLOT X,Y
230  GOTO 200
240  IF A$ = "A" THEN 650
250  IF A$ = "Y" OR A$ = "G" OR A$ = "H" OR
A$ = "B" THEN  GOSUB 730: GOTO 340
```

```
260  IF A$ = "I" OR A$ = "K" OR A$ = "J" OR
A$ = "M" THEN  GOSUB 610: GOTO 340
270  IF A$ = "E" THEN  HOME : TEXT : PR# 3:
PRINT : HOME :: PRINT  CHR$ (18): END
280  IF A$ = "C" THEN 20
290  IF A$ = "T" THEN 380
300  IF A$ = "S" THEN 410
310  IF A$ = "W" THEN 500
320  IF A$ = "L" OR A$ = "O" THEN B$ = A$: G
OTO 350
330  CALL  - 198: GOTO 200
340 C$ = A$
350  IF B$ = "L" THEN  HPLOT X,Y: GOTO 200
360  HPLOT X,Y: FOR I = 1 TO 5: NEXT I: HCOL
OR= 0: HPLOT X,Y
370  HCOLOR= 3: GOTO 200
380  PR# 3: PRINT : HOME : GOSUB 50
390  POKE  - 16300,0: POKE  - 16302,0
400  GOSUB 770: GOTO 200
410  PR# 3: PRINT : HOME
420  INPUT "请输入存盘的文件名:";N$
430  CALL 43089
440 D$ =  CHR$ (4)
450  VTAB 5: HTAB 2: PRINT "正在存盘，请稍等
……"
460  PRINT D$;"BSAVE";N$;",A$2000,L$2000"
470  VTAB 8: HTAB 2: PRINT "存盘结束"
480  FOR I = 1 TO 2000: NEXT I
490  GOTO 390
500  PR# 3: PRINT : HOME
510  VTAB 5: HTAB 2: PRINT "请选择打印方式:1
```

```
一正相;2—放大;3—反相;4—反相放大"
520  GET V$
530  IF V$ = "1" THEN  PR# 1: POKE 1913,1: P
RINT  CHR$ (17): PR# 0: GOTO 580
540  IF V$ = "2" THEN  PR# 1: POKE 1913,65:
PRINT  CHR$ (17): PR# 0: GOTO 580
550  IF V$ = "3" THEN  PR# 1: POKE 1913,33:
PRINT  CHR$ (17): PR# 0: GOTO 580
560  IF V$ = "4" THEN  PR# 1: POKE 1913,97:
PRINT  CHR$ (17): PR# 0: GOTO 580
570  GOTO 430
580  PR# 3: PRINT : VTAB 10: PRINT "硬拷贝完
成"
590  FOR I = 1 TO 1000: NEXT I
600  GOTO 390
610  IF A$ = "I" THEN Y = Y - 1: RETURN
620  IF A$ = "J" THEN X = X - 1: RETURN
630  IF A$ = "K" THEN X = X + 1: RETURN
640  IF A$ = "M" THEN Y = Y + 1: RETURN
650  PR# 3: PRINT : HOME
660  INPUT "请输入调出的文件名:";M$
670  CALL 43089
680  VTAB 5: HTAB 2: PRINT "正在调盘,请稍等
……"
690  PRINT  CHR$ (4)"BLOAD";M$
700  VTAB 8: HTAB 2: PRINT "调盘结束"
710  FOR I = 1 TO 2000: NEXT I
720  GOTO 390
730  IF A$ = "Y" THEN X = X - 1:Y = Y - 1: R
ETURN
740  IF A$ = "G" THEN X = X - 1:Y = Y + 1: R
```

```
ETURN
750  IF A$ = "B" THEN X = X + 1:Y = Y + 1: R
ETURN
760  IF A$ = "H" THEN X = X + 1:Y = Y - 1: R
ETURN
770  HCOLOR= 3: HPLOT 0,0 TO 279,0 TO 279,19
1 TO 0,191 TO 0,0
780  RETURN
```

```
]LIST

10  REM H96
20  POKE 232,106: POKE 233,140
30  SCALE= 1: ROT= 0: HCOLOR= 3
40  HGR2
50 A$ = "CODE":X = 5:Y = 6:W = 7: GOSUB 39
0
60 A$ = "ASCII CHARACTER TABLE":X = 65: GO
SUB 390
70  HPLOT 5,20 TO 270,20: HPLOT 35,5 TO 35
,170
80 A$ = " 32":X = 8:Y = 24: GOSUB 390
90 J = 1:L = 10:X = 60: GOSUB 360
100 A$ = " 42":X = 8:Y = 38: GOSUB 390
110 J = 11:X = 60: GOSUB 360
120 A$ = " 52":X = 8:Y = 52: GOSUB 390
130 J = 21:X = 60: GOSUB 360
140 A$ = " 52":X = 8:Y = 64: GOSUB 390
150 J = 31:X = 60: GOSUB 360
160 A$ = " 62":X = 8:Y = 80: GOSUB 390
170 J = 41:X = 60: GOSUB 360
180 A$ = " 72":X = 8:Y = 94: GOSUB 390
```

```
190 J = 51:X = 60: GOSUB 360
200 A$ = " 82":X = 8:Y = 108: GOSUB 390
210 J = 61:X = 60: GOSUB 360
220 A$ = " 92":X = 8:Y = 122: GOSUB 390
230 J = 71:X = 60: GOSUB 360
240 A$ = "102":X = 8:Y = 136: GOSUB 390
250 J = 81:X = 60: GOSUB 360
260 A$ = "112":X = 8:Y = 150: GOSUB 390
270 J = 91:L = 6:X = 60: GOSUB 360
280  HPLOT 5,170 TO 270,170
290 A$ = "DO YOU WANT TO PRINT (Y/N) ":X =
 60:Y = 175: GOSUB 390
300  GET X$: PRINT D$: IF X$ = "N" OR X$ =
 "" THEN  TEXT : HOME : PRINT "GOODBYE!":
PR# 6
310  IF X$ < > "Y" THEN  PRINT  CHR$ (7):
 GOTO 40
320  HCOLOR= 0:A$ = "DO YOU WANT TO PRINT
(Y/N) ":X = 60:Y = 175: GOSUB 390
330  PR# 1: PRINT  CHR$ (4): POKE 1913,2
340  POKE 1913,2: PRINT  CHR$ (17): PR# 0:
 FOR I = 0 TO 10: PRINT  CHR$ (7): NEXT :
HCOLOR= 3: GOTO 40
350  STOP
360 K = 0: FOR I = J TO J + L - 1
370  DRAW I AT X + 18 * K,Y:K = K + 1: NEX
T
380  RETURN
390  FOR I = 1 TO  LEN (A$)
400  DRAW  ASC ( MID$ (A$,I,1)) - 31 AT X
+ (I - 1) * W,Y: NEXT : RETURN
```

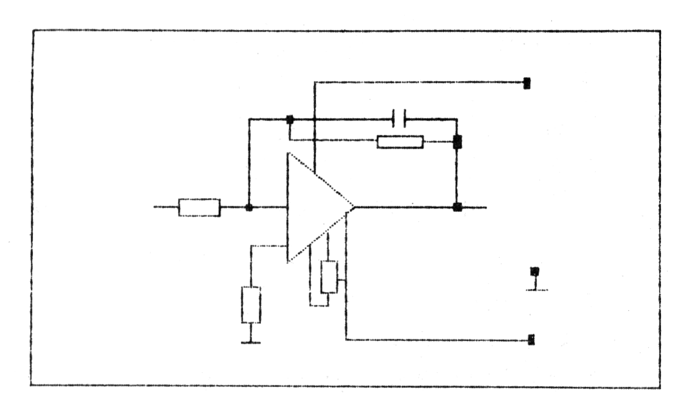


图 9.17

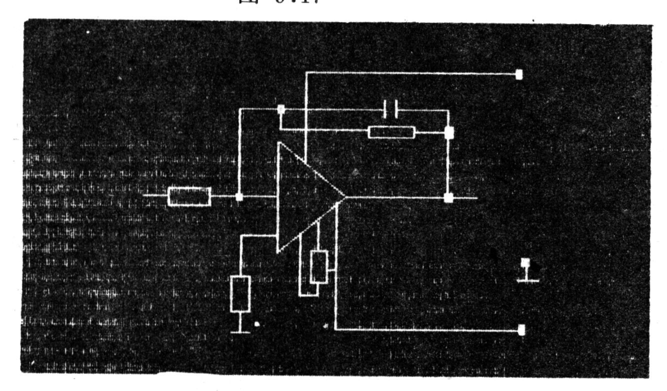


图 9.18

5. 二进制图形表

下面给出的是一个二进制图形表，共有95个造型，它和前面介绍的 ASCII SHAPES 相比，有一个十分显著的特点，即运行速度特别快。

二进制图形表的起始地址：35946($8C6A)，其存贮长度为2454个字节($955)，故图形表的末尾地址是35946+2554=38400($9600)。

程序H96是一个调用二进制图形表的BASIC程序，为了正确实行调用，在程序H96中应安排：

POKE 232,160:POKE 233,140

操作步骤如下：

· CALL—151↙

· 键入二进制图形表的机器语言并存盘，文件名为H97。

· 键入程序H96并存盘。

· 实际使用时，先BLOAD H97↙，后RUN H96↙，立即在屏幕上显示ASCII码造型，若要打印按Y键，中断按CTRL-RESET键。图9.19是一个运动实例。

```
]CALL-151

*8C6A.95FF

8C6A- 61 00 C4 00 D7 00 .
8C70- EA 00 FD 00 10 01 23 01
8C78- 36 01 49 01 5C 01 6F 01
8C80- 82 01 95 01 A8 01 BB 01
8C88- CE 01 E1 01 F4 01 07 02
8C90- 1A 02 2D 02 40 02 53 02
8C98- 66 02 79 02 8C 02 9F 02
8CA0- B2 02 C5 02 D8 02 EB 02
8CA8- FE 02 11 03 24 03 37 03
8CB0- 4A 03 5D 03 70 03 83 03
8CB8- 96 03 A9 03 BC 03 CF 03
8CC0- E2 03 F5 03 08 04 1B 04
8CC8- 2E 04 41 04 54 04 67 04
8CD0- 7A 04 8D 04 A0 04 B3 04
8CD8- C6 04 D9 04 EC 04 FF 04
8CE0- 12 05 25 05 38 05 4B 05
8CE8- 5E 05 71 05 84 05 97 05
8CF0- AA 05 BD 05 D0 05 E3 05
8CF8- F6 05 09 06 1C 06 2F 06
```

```
8D00- 42 06 55 06 68 06 7B 06
8D08- 8E 06 A1 06 B4 06 C7 06
8D10- DA 06 ED 06 00 07 13 07
8D18- 26 07 39 07 4C 07 5F 07
8D20- 72 07 85 07 98 07 AB 07
8D28- BE 07 D1 07 E4 07 09 09
8D30- 1A 1B 13 09 09 1A 1B 13
8D38- 09 09 1A 1B 13 09 09 1A
8D40- 00 09 0D 1A 3B 13 09 0D
8D48- 1A 3B 13 09 09 1A 1B 13
8D50- 09 0D 1A 00 29 29 1A 1F
8D58- 17 29 29 1A 1B 13 09 09
8D60- 1A 1B 13 09 09 1A 00 29
8D68- 29 1A 1F 17 2D 2D 1E 1F
8D70- 17 2D 2D 1E 1F 17 29 29
8D78- 1A 00 09 0D 3A 3F 17 0D
8D80- 0D 3A 3F 17 09 0D 1E 3F
8D88- 37 09 0D 1A 00 2D 09 1E
8D90- 1F 37 09 0D 1A 1B 17 0D
8D98- 09 3A 1F 13 09 29 1E 00
8DA0- 29 09 1A 3B 33 0D 0D 3A
8DA8- 1B 17 0D 0D 1E 1F 33 29
8DB0- 0D 1E 00 09 29 1A 3B 13
8DB8- 29 09 1A 1B 13 09 09 1A
8DC0- 1B 13 09 09 1A 00 09 29
8DC8- 1A 3B 13 29 09 1A 1B 17
8DD0- 29 09 1A 3B 13 09 29 1A
8DD8- 00 29 09 1A 3B 13 09 29
8DE0- 1A 1F 13 09 29 1A 3B 13
8DE8- 29 09 1A 00 09 0D 3A 3B
8DF0- 33 29 2D 1A 3B 13 29 2D
8DF8- 3A 3B 33 09 0D 1A 00 09
8E00- 09 1A 3B 13 09 0D 3A 3F
8E08- 37 09 0D 1A 3B 13 09 09
8E10- 1A 00 09 09 1A 1B 13 09
8E18- 09 1A 1B 13 29 0D 1A 3B
8E20- 13 29 09 1A 00 09 09 1A
8E28- 1B 13 09 09 3A 3F 37 09
8E30- 09 1A 1B 13 09 09 1A 00
8E38- 09 09 1A 1B 13 09 09 1A
8E40- 1B 13 09 09 1A 1B 13 09
8E48- 0D 1A 00 09 09 3A 1B 13
8E50- 09 29 1A 3B 13 29 09 1A
8E58- 1B 33 09 09 1A 00 29 2D
8E60- 3A 1B 33 0D 29 3E 3B 33
8E68- 2D 09 3E 1B 33 29 2D 1A
8E70- 00 09 0D 1A 3B 17 09 0D
```

```
8E78- 1A 3B 13 09 0D 1A 3B 13
8E80- 29 2D 1A 00 29 2D 3A 1B
8E88- 33 09 09 1E 1F 13 09 0D
8E90- 1A 1B 17 2D 2D 1E 00 29
8E98- 2D 3A 1B 33 09 09 1E 3F
8EA0- 13 09 09 3E 1B 33 29 2D
8EA8- 1A 00 09 29 1A 3F 13 29
8EB0- 29 1A 1F 33 2D 2D 1E 1F
8EB8- 13 09 29 1A 00 2D 2D 1E
8EC0- 1B 33 2D 2D 3A 1B 13 09
8EC8- 09 3E 1B 33 29 2D 1A 00
8ED0- 09 2D 1E 1B 17 0D 09 1A
8ED8- 3F 37 0D 09 3E 1B 33 29
8EE0- 2D 1A 00 2D 2D 3E 1B 13
8EE8- 09 29 1A 3B 13 29 09 1A
8EF0- 1B 17 29 09 1A 00 29 2D
8EF8- 3A 1B 33 0D 09 1E 3F 17
8F00- 0D 09 3E 1B 33 29 2D 1A
8F08- 00 29 2D 3A 1B 33 0D 09
8F10- 3E 3F 17 09 09 1E 1F 13
8F18- 2D 0D 1A 00 09 09 1A 1B
8F20- 13 09 0D 1A 1B 13 09 0D
8F28- 1A 1B 13 09 09 1A 00 09
8F30- 09 1A 1B 13 09 0D 1A 1B
8F38- 13 09 0D 1A 3B 13 29 09
8F40- 1A 00 09 29 1A 3B 13 29
8F48- 09 1A 1B 33 29 09 1A 3B
8F50- 13 09 29 1A 00 09 09 1A
8F58- 1B 13 2D 2D 1E 1B 13 2D
8F60- 2D 1E 1B 13 09 09 1A 00
8F68- 29 09 1A 3B 13 09 29 3A
8F70- 1B 13 09 29 1A 3B 13 29
8F78- 09 1A 00 29 2D 3A 1B 33
8F80- 09 09 1E 1F 13 09 0D 1A
8F88- 1B 13 09 0D 1A 00 29 2D
8F90- 3A 1B 33 0D 0D 3E 3F 33
8F98- 0D 2D 1A 1B 33 29 2D 1E
8FA0- 00 09 0D 1A 1F 17 0D 09
8FA8- 3E 1B 33 2D 2D 3E 1B 33
8FB0- 0D 09 1E 00 2D 2D 3A 1B
8FB8- 33 0D 09 1E 3F 37 0D 09
8FC0- 3E 1B 33 2D 2D 1A 00 29
8FC8- 2D 3A 1B 33 0D 09 1A 1B
8FD0- 33 0D 09 3A 1B 33 29 2D
8FD8- 1A 00 2D 2D 3A 1B 33 0D
8FE0- 09 3E 1B 33 0D 09 3E 1B
8FE8- 33 2D 2D 1A 00 2D 2D 1E
```

```
8FF0-  1B 33 0D 09 1A 3F 37 0D
8FF8-  09 1A 1B 33 2D 2D 1E 00
9000-  2D 2D 1E 1B 33 0D 09 1A
9008-  3F 37 0D 09 1A 1B 33 0D
9010-  09 1A 00 29 2D 1E 1B 33
9018-  0D 09 1A 1B 33 0D 29 3E
9020-  1B 33 29 2D 1E 00 0D 09
9028-  3E 1B 33 0D 09 3E 3F 37
9030-  0D 09 3E 1B 33 0D 09 1E
9038-  00 29 2D 1A 3B 13 09 0D
9040-  1A 3B 13 09 0D 1A 3B 13
9048-  29 2D 1A 00 09 09 3E 1B
9050-  13 09 09 3E 1B 13 09 09
9058-  3E 1B 33 29 2D 1A 00 0D
9060-  09 1E 1F 33 0D 0D 1A 1B
9068-  37 0D 0D 1A 1F 33 0D 09
9070-  1E 00 0D 09 1A 1B 33 0D
9078-  09 1A 1B 33 0D 09 1A 1B
9080-  33 2D 2D 1E 00 0D 09 3E
9088-  1F 37 0D 0D 3E 3B 33 0D
9090-  09 3E 1B 33 0D 09 1E 00
9098-  0D 09 3E 1B 33 2D 09 3E
90A0-  3B 33 0D 29 3E 1B 33 0D
90A8-  09 1E 00 29 2D 3A 1B 33
90B0-  0D 09 3E 1B 33 0D 09 3E
90B8-  1B 33 29 2D 1A 00 2D 2D
90C0-  3A 1B 33 0D 09 1E 3F 37
90C8-  0D 09 1A 1B 33 0D 09 1A
90D0-  00 29 2D 3A 1B 33 0D 09
90D8-  3E 1B 33 0D 0D 1E 1F 33
90E0-  29 0D 1E 00 2D 2D 3A 1B
90E8-  33 0D 09 1E 3F 37 0D 0D
90F0-  1A 1F 33 0D 09 1E 00 29
90F8-  2D 3A 1B 33 0D 09 1A 3F
9100-  17 09 09 3E 1B 33 29 2D
9108-  1A 00 2D 2D 1E 3B 13 09
9110-  0D 1A 3B 13 09 0D 1A 3B
9118-  13 09 0D 1A 00 0D 09 3E
9120-  1B 33 0D 09 3E 1B 33 0D
9128-  09 3E 1B 33 29 2D 1A 00
9130-  0D 09 3E 1B 33 0D 09 3E
9138-  1B 33 0D 09 1E 1F 17 09
9140-  0D 1A 00 0D 09 3E 1B 33
9148-  0D 09 3E 3B 33 0D 0D 3E
9150-  1F 37 0D 09 1E 00 0D 09
9158-  3E 1B 33 29 29 1A 3B 13
9160-  29 29 3A 1B 33 0D 09 1E
```

```
9168- 00 0D 09 3E 1B 33 29 29
9170- 1A 3B 13 09 0D 1A 3B 13
9178- 09 0D 1A 00 2D 2D 3E 1B
9180- 13 09 29 1A 3B 13 29 09
9188- 1A 1B 33 2D 2D 1E 00 29
9190- 2D 1A 1B 17 29 09 1A 1B
9198- 17 29 09 1A 1B 17 29 2D
91A0- 1A 00 09 09 1A 1B 33 29
91A8- 09 1A 3B 13 09 29 3A 1B
91B0- 13 09 09 1A 00 29 2D 1A
91B8- 1F 13 09 29 1A 1F 13 09
91C0- 29 1A 1F 13 29 2D 1A 00
91C8- 09 0D 1A 1F 17 0D 09 1E
91D0- 1B 13 09 09 1A 1B 13 09
91D8- 09 1A 00 09 09 1A 1B 13
91E0- 09 09 1A 1B 13 09 09 1A
91E8- 1B 13 2D 2D 1E 00 29 09
91F0- 1A 3B 13 09 29 1A 1B 13
91F8- 09 09 1A 1B 13 09 09 1A
9200- 39 39 31 31 42 42 38 38
9208- 2D 2D 20 20 32 32 39 39
9210- 20 20 30 30 44 44 20 20
9218- 30 30 39 39 20 20 30 30
9220- 39 39 20 20 30 30 39 39
9228- 20 20 30 30 39 39 20 20
9230- 32 32 39 39 20 20 30 30
9238- 33 33 80 20 80 20 80 20
9240- 80 20 80 20 39 39 32 32
9248- 30 30 38 38 2D 2D 20 20
9250- 32 32 30 30 20 20 32 32
9258- 30 30 20 20 33 33 39 39
9260- 20 20 32 32 30 30 20 20
9268- 33 33 39 39 20 20 33 33
9270- 30 30 20 20 33 33 30 30
9278- 20 20 32 32 30 30 80 20
9280- 80 20 80 20 80 20 80 20
9288- 39 39 32 32 35 35 30 30
9290- 2D 2D 20 20 33 33 30 30
9298- 20 20 32 32 30 30 20 20
92A0- 33 33 39 39 20 20 33 33
92A8- 30 30 20 20 33 33 33 33
92B0- 20 20 33 33 30 30 20 20
92B8- 38 38 30 30 20 20 32 32
92C0- 30 30 80 20 80 20 80 20
92C8- 80 20 80 20 39 39 32 32
92D0- 41 41 30 30 2D 2D 20 20
92D8- 33 33 30 30 20 20 33 33
```

```
92E0- 30 30 20 20 33 33 33 33
92E8- 20 20 33 33 30 30 20 20
92F0- 38 38 30 30 20 20 32 32
92F8- 39 39 20 20 33 33 32 32
9300- 20 20 32 32 30 30 80 20
9308- 80 20 80 20 80 20 80 20
9310- 39 39 32 32 45 45 38 38
9318- 2D 2D 20 20 33 33 38 38
9320- 20 20 33 33 39 39 20 20
9328- 33 33 32 32 20 20 33 33
9330- 30 30 20 20 38 38 30 30
9338- 20 20 34 34 35 35 20 20
9340- 33 33 38 38 20 20 33 33
9348- 30 30 80 20 80 20 80 20
9350- 80 20 80 20 39 39 33 33
9358- 33 33 38 38 2D 2D 20 20
9360- 33 33 33 33 20 20 33 33
9368- 30 30 20 20 38 38 30 30
9370- 20 20 32 32 30 30 20 20
9378- 32 32 44 44 20 20 32 32
9380- 30 30 20 20 33 33 33 33
9388- 20 20 33 33 30 30 80 20
9390- 80 20 80 20 80 20 80 20
9398- 39 39 33 33 38 38 30 30
93A0- 2D 2D 20 20 32 32 30 30
93A8- 20 20 32 32 30 30 20 20
93B0- 33 33 33 33 20 20 33 33
93B8- 30 30 20 20 33 33 32 32
93C0- 20 20 33 33 30 30 20 20
93C8- 33 33 33 33 20 20 33 33
93D0- 32 32 80 20 80 20 80 20
93D8- 80 20 80 20 39 39 33 33
93E0- 44 44 30 30 2D 2D 20 20
93E8- 38 38 30 30 20 20 32 32
93F0- 34 34 20 20 33 33 30 30
93F8- 20 20 33 33 30 30 20 20
9400- 33 33 33 33 20 20 33 33
9408- 30 30 20 20 32 32 30 30
9410- 20 20 33 33 33 33 80 20
9418- 80 20 80 20 80 20 80 20
9420- 39 39 34 34 31 31 38 38
9428- 2D 2D 20 20 33 33 39 39
9430- 20 20 32 32 44 44 20 20
9438- 33 33 32 32 20 20 33 33
9440- 32 32 20 20 32 32 30 30
9448- 20 20 33 33 32 32 20 20
```

```
9450- 32 32 30 30 20 20 32 32
9458- 32 32 80 20 80 20 80 20
9460- 80 20 80 20 39 39 34 34
9468- 36 36 38 38 2D 2D 20 20
9470- 80 38 30 30 20 20 33 33
9478- 80 20 20 20 32 32 30 30
9480- 80 20 80 20 30 30 20 20
9488- 80 20 80 20 80 20 32 32
9490- 80 20 80 20 80 20 80 30
9498- 80 20 80 20 80 20 80 20
94A0- 80 20 80 20 80 20 80 20
94A8- 80 20 80 20 80 20 80 20
94B0- 80 20 80 20 80 20 80 20
94B8- 80 20 80 20 80 20 80 20
94C0- 80 20 80 20 80 20 80 20
94C8- 80 20 80 20 80 20 80 20
94D0- 00 00 00 00 00 00 00 00
94D8- 00 00 00 00 00 00 00 00
94E0- 00 00 00 00 00 00 00 00
94E8- 00 00 00 00 00 00 00 00
94F0- 00 00 53 F6 FF 01 1D 00
94F8- 09 1D 09 1D 09 00 C8 CE
9500- FF 95 F0 03 4C D9 94 CA
9508- CA A9 00 9D 4E 94 60 00
9510- A9 09 85 08 06 09 90 06
9518- A9 04 05 08 85 08 06 09
9520- 90 04 A9 20 05 08 9D 4E
9528- 94 E8 60 85 08 46 09 90
9530- 06 A9 04 05 08 85 08 46
9538- 09 90 04 A9 20 05 08 9D
9540- 4E 94 E8 60 A9 1A 85 08
9548- 06 09 90 06 A9 04 05 08
9550- 85 08 60 46 09 90 06 A9
9558- 20 05 08 85 08 A5 08 9D
9560- 4E 94 E8 60 A2 00 A0 00
9568- A9 07 8D FF 95 B1 06 85
9570- 09 06 09 06 09 20 10 95
9578- 20 10 95 20 10 95 C8 B1
9580- 06 85 09 06 09 06 09 20
9588- 10 95 20 10 95 20 10 95
9590- C8 B1 06 85 09 06 09 06
9598- 09 06 09 06 09 06 09 20
95A0- 10 95 20 44 95 C8 C8 C8
95A8- B1 06 85 09 20 53 95 A9
95B0- 1B 20 2B 95 88 B1 06 85
95B8- 09 A9 1B 20 2B 95 A9 1B
95C0- 20 2B 95 A9 1B 20 2B 95
```

```
95C8- 88 B1 06 85 09 A9 1B 20
95D0- 2B 95 A9 1B 20 2B 95 A9
95D8- 13 20 2B 95 C8 C8 C8 CE
95E0- FF 95 F0 03 4C 6D 95 A9
95E8- 00 9D 4E 94 60 A2 6A BD
95F0- 4D 94 9D 3A 94 CA D0 F7
95F8- 60 A2 13 4C EF 95 00 20
```

CODE	ASCII CHARACTER TABLE									
32		!	"	#	$	%	&	'	(	)
42	*	+	,	-	.	/	0	1	2	3
52	4	5	6	7	8	9	:	;	<	=
52	>	?	@	A	B	C	D	E	F	G
62	H	I	J	K	L	M	N	O	P	Q
72	R	S	T	U	V	W	X	Y	Z	[
82	\	]	^	_	`	a	b	c	d	e
92	f	g	h	i	j	k	l	m	n	o
102	p	q	r	s	t	u	v	w	x	y
112	z	{	|	}	~	■				

图 9.19

附录一、中华学习机字符区位码

第 1 区

0101	0102 、	0103 。	0104 ·	0105 ˉ
0106 ˇ	0107 ¨	0108 〃	0109 々	0110 —
0111 ～	0112 ‖	0113 …	0114 ‘	0115 ’
0116 “	0117 ”	0118 〔	0119 〕	0120 〈
0121 〉	0122 《	0123 》	0124 「	0125 」
0126 『	0127 』	0128 〖	0129 〗	0130 【
0131 】	0132 ±	0133 ×	0134 ÷	0135 ∶
0136 ∧	0137 ∨	0138 ∑	0139 ∏	0140 ∪
0141 ∩	0142 ∈	0143 ∷	0144 √	0145 ⊥
0146 ∥	0147 ∠	0148 ⌒	0149 ⊙	0150 ∫
0151 ∮	0152 ≡	0153 ≌	0154 ≈	0155 ∽
0156 ∝	0157 ≠	0158 ≮	0159 ≯	0160 ＜
0161 ＞	0162 ∞	0163 ∵	0164 ∴	0165 ♂
0166 ♀	0167 °	0168 ′	0169 ″	0170 ℃
0171 ＄	0172 ¤	0173 ￠	0174 ￡	0175 ‰
0176 §	0177 №	0178 ☆	0179 ★	0180 ○
0181 ●	0182 ◎	0183 ◇	0184 ◆	0185 □
0186 ■	0187 △	0188 ▲	0189 ※	0190 →
0191 ←	0192 ↑	0193 ↓	0194 〓	

第 2 区

0201	0202	0203	0204	0205
0206	0207	0208	0209	0210

0211		0212		0213		0214		0215	
0216		0217	⒈	0218	⒉	0219	⒊	0220	⒋
0221	⒌	0222	⒍	0223	⒎	0224	⒏	0225	⒐
0226	⒑	0227	⒒	0228	⒓	0229	⒔	0230	⒕
0231	⒖	0232	⒗	0233	⒘	0234	⒙	0235	⒚
0236	⒛	0237	⑴	0238	⑵	0239	⑶	0240	⑷
0241	⑸	0242	⑹	0243	⑺	0244	⑻	0245	⑼
0246	⑽	0247	⑾	0248	⑿	0249	⒀	0250	⒁
0251	⒂	0252	⒃	0253	⒄	0254	⒅	0255	⒆
0256	⒇	0257	①	0258	②	0259	③	0260	④
0261	⑤	0262	⑥	0263	⑦	0264	⑧	0265	⑨
0266	⑩	0267		0268		0269	㈠	0270	㈡
0271	㈢	0272	㈣	0273	㈤	0274	㈥	0275	㈦
0276	㈧	0277	㈨	0278	㈩	0279		0280	
0281	Ⅰ	0282	Ⅱ	0283	Ⅲ	0284	Ⅳ	0285	Ⅴ
0286	Ⅵ	0287	Ⅶ	0288	Ⅷ	0289	Ⅸ	0290	Ⅹ
0291	Ⅺ	0292	Ⅻ	0293		0294			

第 3 区

0301	！	0302	＂	0303	＃	0304	￥	0305	％
0306	＆	0307	＇	0308	（	0309	）	0310	＊
0311	＋	0312	，	0313	－	0314	．	0315	／
0316	０	0317	１	0318	２	0319	３	0320	４
0321	５	0322	６	0323	７	0324	８	0325	９
0326	：	0327	；	0328	＜	0329	＝	0330	＞
0331	？	0332	＠	0333	Ａ	0334	Ｂ	0335	Ｃ
0336	Ｄ	0337	Ｅ	0338	Ｆ	0339	Ｇ	0340	Ｈ
0341	Ｉ	0342	Ｊ	0343	Ｋ	0344	Ｌ	0345	Ｍ

0346 N 0347 O 0348 P 0349 Q 0350 R
0351 S 0352 T 0353 U 0354 V 0355 W
0356 X 0357 Y 0358 Z 0359 [0360 \
0361] 0362 ^ 0363 _ 0364 ` 0365 a
0366 b 0367 c 0368 d 0369 e 0370 f
0371 g 0372 h 0373 i 0374 j 0375 k
0376 l 0377 m 0378 n 0379 o 0380 p
0381 q 0382 r 0383 s 0384 t 0385 u
0386 v 0387 w 0388 x 0389 y 0390 z
0391 { 0392 | 0393 } 0394 ‾

第 4 区

0401 ぁ 0402 あ 0403 ぃ 0404 い 0405 ぅ
0406 う 0407 ぇ 0408 え 0409 ぉ 0410 お
0411 か 0412 が 0413 き 0414 ぎ 0415 く
0416 ぐ 0417 け 0418 げ 0419 こ 0420 ご
0421 さ 0422 ざ 0423 し 0424 じ 0425 す
0426 ず 0427 せ 0428 ぜ 0429 そ 0430 ぞ
0431 た 0432 だ 0433 ち 0434 ぢ 0435 っ
0436 つ 0437 づ 0438 て 0439 で 0440 と
0441 ど 0442 な 0443 に 0444 ぬ 0445 ね
0446 の 0447 は 0448 ば 0449 ぱ 0450 ひ
0451 び 0452 ぴ 0453 ふ 0454 ぶ 0455 ぷ
0456 へ 0457 べ 0458 ぺ 0459 ほ 0460 ぼ
0461 ぽ 0462 ま 0463 み 0464 む 0465 め
0466 も 0467 ゃ 0468 や 0469 ゅ 0470 ゆ
0471 ょ 0472 よ 0473 ら 0474 り 0475 る

0476 れ　0477 ろ　0478 ゎ　0479 わ　0480 ゐ
0481 ゑ　0482 を　0483 ん　0484　0485
0486　0487　0488　0489　0490
0491　0492　0493　0494

第 5 区

0501 ァ　0502 ア　0503 ィ　0504 イ　0505 ゥ
0506 ウ　0507 ェ　0508 エ　0509 ォ　0510 オ
0511 カ　0512 ガ　0513 キ　0514 ギ　0515 ク
0516 グ　0517 ケ　0518 ゲ　0519 コ　0520 ゴ
0521 サ　0522 ザ　0523 シ　0524 ジ　0525 ス
0526 ズ　0527 セ　0528 ゼ　0529 ソ　0530 ゾ
0531 タ　0532 ダ　0533 チ　0534 ヂ　0535 ッ
0536 ツ　0537 ヅ　0538 テ　0539 デ　0540 ト
0541 ド　0542 ナ　0543 ニ　0544 ヌ　0545 ネ
0546 ノ　0547 ハ　0548 バ　0549 パ　0550 ヒ
0551 ビ　0552 ピ　0553 フ　0554 ブ　0555 プ
0556 ヘ　0557 ベ　0558 ペ　0559 ホ　0560 ボ
0561 ポ　0562 マ　0563 ミ　0564 ム　0565 メ
0566 モ　0567 ャ　0568 ヤ　0569 ュ　0570 ユ
0571 ョ　0572 ヨ　0573 ラ　0574 リ　0575 ル
0576 レ　0577 ロ　0578 ヮ　0579 ワ　0580 ヰ
0581 ヱ　0582 ヲ　0583 ン　0584 ヴ　0585 ヵ
0586 ヶ　0587　0588　0589　0590
0591　0592　0593　0594

第 6 区

0601	Α	0602	Β	0603	Γ	0604	Δ	0605	Ε
0606	Ζ	0607	Η	0608	Θ	0609	Ι	0610	Κ
0611	Λ	0612	Μ	0613	Ν	0614	Ξ	0615	Ο
0616	Π	0617	Ρ	0618	Σ	0619	Τ	0620	Υ
0621	Φ	0622	Χ	0623	Ψ	0624	Ω	0625	
0626		0627		0628		0629		0630	
0631		0632		0633	α	0634	β	0635	γ
0636	δ	0637	ε	0638	ζ	0639	η	0640	θ
0641	ι	0642	κ	0643	λ	0644	μ	0645	ν
0646	ξ	0647	ο	0648	π	0649	ρ	0650	σ
0651	τ	0652	υ	0653	φ	0654	χ	0655	ψ
0656	ω	0657		0658		0659		0660	
0661		0662		0663		0664		0665	
0666		0667		0668		0669		0670	
0671		0672		0673		0674		0675	
0676		0677		0678		0679		0680	
0681		0682		0683		0684		0685	
0686		0687		0688		0689		0690	
0691		0692		0693		0694			

第 7 区

0701	А	0702	Б	0703	В	0704	Г	0705	Д
0706	Е	0707	Ё	0708	Ж	0709	З	0710	И
0711	Й	0712	К	0713	Л	0714	М	0715	Н
0716	О	0717	П	0718	Р	0719	С	0720	Т

0721	У	0722	Х	0723	Ф	0724	Ц	0725	Ч
0726	Ш	0727	Щ	0728	Ъ	0729	Ы	0730	Ь
0731	Э	0732	Ю	0733	Я	0734		0735	
0736		0737		0738		0739		0740	
0741		0742		0743		0744		0745	
0746		0747		0748		0749	а	0750	б
0751	в	0752	г	0753	д	0754	е	0755	ё
0756	ж	0757	з	0758	и	0759	й	0760	к
0761	л	0762	м	0763	н	0764	о	0765	п
0766	р	0767	с	0768	т	0769	у	0770	ф
0771	х	0772	ц	0773	ч	0774	ш	0775	щ
0776	ъ	0777	ы	0778	ь	0779	э	0780	ю
0781	я	0782		0783		0784		0785	
0786		0787		0788		0789		0790	
0791		0792		0793		0794			

第 8 区

0801	ā	0802	á	0803	ǎ	0804	à	0805	ē
0806	é	0807	ě	0808	è	0809	ī	0810	í
0811	ǐ	0812	ì	0813	ō	0814	ó	0815	ǒ
0816	ò	0817	ū	0818	ú	0819	ǔ	0820	ù
0821	ǖ	0822	ǘ	0823	ǚ	0824	ǜ	0825	ü
0826	ê	0827	ɑ	0828	ɡ	0829	ḿ	0830	ń
0831	ň	0832	ǹ	0833		0834		0835	
0836		0837	ㄅ	0838	ㄆ	0839	ㄇ	0840	ㄈ
0841	ㄉ	0842	ㄊ	0843	ㄋ	0844	ㄌ	0845	ㄍ
0846	ㄎ	0847	ㄏ	0848	ㄐ	0849	ㄑ	0850	ㄒ

0851	ㄓ	0852	ㄔ	0853	ㄕ	0854	ㄖ	0855	ㄗ
0856	ㄘ	0857	ㄙ	0858	ㄚ	0859	ㄛ	0860	ㄜ
0861	ㄝ	0862	ㄞ	0863	ㄟ	0864	ㄠ	0865	ㄡ
0866	ㄢ	0867	ㄣ	0868	ㄤ	0869	ㄥ	0870	ㄦ
0871	ㄧ	0872	ㄨ	0873	ㄩ	0874		0875	
0876		0877		0878		0879		0880	
0881		0882		0883		0884		0885	
0886		0887		0888		0889		0890	
0891		0892		0893		0894			

第 9 区

0901		0902		0903		0904	─	0905	━
0906	│	0907	┃	0908	┄	0909	┅	0910	┆
0911	┇	0912	┈	0913	┉	0914	┊	0915	┋
0916	┌	0917	┍	0918	┎	0919	┏	0920	┐
0921	┑	0922	┒	0923	┓	0924	└	0925	┕
0926	┖	0927	┗	0928	┘	0929	┙	0930	┚
0931	┛	0932	├	0933	┝	0934	┞	0935	┟
0936	┠	0937	┡	0938	┢	0939	┣	0940	┤
0941	┥	0942	┦	0943	┧	0944	┨	0945	┩
0946	┪	0947	┫	0948	┬	0949	┭	0950	┮
0951	┯	0952	┰	0953	┱	0954	┲	0955	┳
0956	┴	0957	┵	0958	┶	0959	┷	0960	┸
0961	┹	0962	┺	0963	┻	0964	┼	0965	┽
0966	┾	0967	┿	0968	╀	0969	╁	0970	╂
0971	╃	0972	╄	0973	╅	0974	╆	0975	╇
0976	╈	0977	╉	0978	╊	0979	╋	0980	

0981	0982	0983	0984	0985
0986	0987	0988	0989	0990
0991	0992	0993	0994	

附录二、中华学习机汉字区位码

第 16 区

1601 啊　1602 阿　1603 埃　1604 挨　1605 哎
1606 唉　1607 哀　1608 皑　1609 癌　1610 蔼
1611 矮　1612 艾　1613 碍　1614 爱　1615 隘
1616 鞍　1617 氨　1618 安　1619 俺　1620 按
1621 暗　1622 岸　1623 胺　1624 案　1625 肮
1626 昂　1627 盎　1628 凹　1629 敖　1630 熬
1631 翱　1632 袄　1633 傲　1634 奥　1635 懊
1636 澳　1637 芭　1638 捌　1639 扒　1640 叭
1641 吧　1642 笆　1643 八　1644 疤　1645 巴
1646 拔　1647 跋　1648 靶　1649 把　1650 耙
1651 坝　1652 霸　1653 罢　1654 爸　1655 白
1656 柏　1657 百　1658 摆　1659 佰　1660 败
1661 拜　1662 稗　1663 斑　1664 班　1665 搬
1666 扳　1667 般　1668 颁　1669 板　1670 版
1671 扮　1672 拌　1673 伴　1674 瓣　1675 半
1676 办　1677 绊　1678 邦　1679 帮　1680 梆
1681 榜　1682 膀　1683 绑　1684 棒　1685 磅
1686 蚌　1687 镑　1688 傍　1689 谤　1690 苞
1691 胞　1692 包　1693 褒　1694 剥

第 17 区

1701 薄　1702 雹　1703 保　1704 堡　1705 饱
1706 宝　1707 抱　1708 报　1709 暴　1710 豹

1711 鲍　1712 爆　1713 杯　1714 碑　1715 悲
1716 卑　1717 北　1718 辈　1719 背　1720 贝
1721 钡　1722 倍　1723 狈　1724 备　1725 惫
1726 焙　1727 被　1728 奔　1729 苯　1730 本
1731 笨　1732 崩　1733 绷　1734 甭　1735 泵
1736 蹦　1737 迸　1738 逼　1739 鼻　1740 比
1741 鄙　1742 笔　1743 彼　1744 碧　1745 蓖
1746 蔽　1747 毕　1748 毙　1749 毖　1750 币
1751 庇　1752 痹　1753 闭　1754 敝　1755 弊
1756 必　1757 辟　1758 壁　1759 臂　1760 避
1761 陛　1762 鞭　1763 边　1764 编　1765 贬
1766 扁　1767 便　1768 变　1769 卞　1770 辨
1771 辩　1772 辫　1773 遍　1774 标　1775 彪
1776 膘　1777 表　1778 鳖　1779 憋　1780 别
1781 瘪　1782 彬　1783 斌　1784 濒　1785 滨
1786 宾　1787 摈　1788 兵　1789 冰　1790 柄
1791 丙　1792 秉　1793 饼　1794 炳

第 18 区

1801 病　1802 并　1803 玻　1804 菠　1805 播
1806 拨　1807 钵　1808 波　1809 博　1810 勃
1811 搏　1812 铂　1813 箔　1814 伯　1815 帛
1816 舶　1817 脖　1818 膊　1819 渤　1820 泊
1821 驳　1822 捕　1823 卜　1824 哺　1825 补
1826 埠　1827 不　1828 布　1829 步　1830 簿
1831 部　1832 怖　1833 擦　1834 猜　1835 裁
1836 材　1837 才　1838 财　1839 睬　1840 踩
1841 采　1842 彩　1843 菜　1844 蔡　1845 餐

1846 参	1847 蚕	1848 残	1849 惭	1850 惨
1851 灿	1852 苍	1853 舱	1854 仓	1855 沧
1856 藏	1857 操	1858 糙	1859 槽	1860 曹
1861 草	1862 厕	1863 策	1864 侧	1865 册
1866 测	1867 层	1868 蹭	1869 插	1870 叉
1871 茬	1872 茶	1873 查	1874 碴	1875 搽
1876 察	1877 岔	1878 差	1879 诧	1880 拆
1881 柴	1882 豺	1883 搀	1884 掺	1885 蝉
1886 馋	1887 谗	1888 缠	1889 铲	1890 产
1891 阐	1892 颤	1893 昌	1894 猖	

第 19 区

1901 场	1902 尝	1903 常	1904 长	1905 偿
1906 肠	1907 厂	1908 敞	1909 畅	1910 唱
1911 倡	1912 超	1913 抄	1914 钞	1915 朝
1916 嘲	1917 潮	1918 巢	1919 吵	1920 炒
1921 车	1922 扯	1923 撤	1924 掣	1925 彻
1926 澈	1927 郴	1928 臣	1929 辰	1930 尘
1931 晨	1932 忱	1933 沉	1934 陈	1935 趁
1936 衬	1937 撑	1938 称	1939 城	1940 橙
1941 成	1942 呈	1943 乘	1944 程	1945 惩
1946 澄	1947 诚	1948 承	1949 逞	1950 骋
1951 秤	1952 吃	1953 痴	1954 持	1955 匙
1956 池	1957 迟	1958 弛	1959 驰	1960 耻
1961 齿	1962 侈	1963 尺	1964 赤	1965 翅
1966 斥	1967 炽	1968 充	1969 冲	1970 虫
1971 崇	1972 宠	1973 抽	1974 酬	1975 畴
1976 踌	1977 稠	1978 愁	1979 筹	1980 仇

1981 蛔 1982 瞅 1983 丑 1984 臭 1985 初
1986 出 1987 橱 1988 厨 1989 躇 1990 锄
1991 雏 1992 滁 1993 除 1994 楚

第 20 区

2001 础 2002 储 2003 矗 2004 搐 2005 触
2006 处 2007 揣 2008 川 2009 穿 2010 椽
2011 传 2012 船 2013 喘 2014 串 2015 疮
2016 窗 2017 幢 2018 床 2019 闯 2020 创
2021 吹 2022 炊 2023 捶 2024 锤 2025 垂
2026 春 2027 椿 2028 醇 2029 唇 2030 淳
2031 纯 2032 蠢 2033 戳 2034 绰 2035 疵
2036 茨 2037 磁 2038 雌 2039 辞 2040 慈
2041 瓷 2042 词 2043 此 2044 刺 2045 赐
2046 次 2047 聪 2048 葱 2049 囱 2050 匆
2051 从 2052 丛 2053 凑 2054 粗 2055 醋
2056 簇 2057 促 2058 蹿 2059 篡 2060 窜
2061 摧 2062 崔 2063 催 2064 脆 2065 瘁
2066 粹 2067 淬 2068 翠 2069 村 2070 存
2071 寸 2072 磋 2073 撮 2074 搓 2075 措
2076 挫 2077 错 2078 搭 2079 达 2080 答
2081 瘩 2082 打 2083 大 2084 呆 2085 歹
2086 傣 2087 戴 2088 带 2089 殆 2090 代
2091 贷 2092 袋 2093 待 2094 逮

第 21 区

2101 怠 2102 耽 2103 担 2104 丹 2105 单
2106 郸 2107 掸 2108 胆 2109 旦 2110 氮

2111 但	2112 惮	2113 淡	2114 诞	2115 弹
2116 蛋	2117 当	2118 挡	2119 党	2120 荡
2121 档	2122 刀	2123 捣	2124 蹈	2125 倒
2126 岛	2127 祷	2128 导	2129 到	2130 稻
2131 悼	2132 道	2133 盗	2134 德	2135 得
2136 的	2137 蹬	2138 灯	2139 登	2140 等
2141 瞪	2142 凳	2143 邓	2144 堤	2145 低
2146 滴	2147 迪	2148 敌	2149 笛	2150 狄
2151 涤	2152 翟	2153 嫡	2154 抵	2155 底
2156 地	2157 蒂	2158 第	2159 帝	2160 弟
2161 递	2162 缔	2163 颠	2164 掂	2165 滇
2166 碘	2167 点	2168 典	2169 靛	2170 垫
2171 电	2172 佃	2173 甸	2174 店	2175 惦
2176 奠	2177 淀	2178 殿	2179 碉	2180 叼
2181 雕	2182 凋	2183 刁	2184 掉	2185 吊
2186 钓	2187 调	2188 跌	2189 爹	2190 碟
2191 蝶	2192 迭	2193 谍	2194 叠	

第 22 区

2201 丁	2202 盯	2203 叮	2204 钉	2205 顶
2206 鼎	2207 锭	2208 定	2209 订	2210 丢
2211 东	2212 冬	2213 董	2214 懂	2215 动
2216 栋	2217 侗	2218 恫	2219 冻	2220 洞
2221 兜	2222 抖	2223 斗	2224 陡	2225 豆
2226 逗	2227 痘	2228 都	2229 督	2230 毒
2231 犊	2232 独	2233 读	2234 堵	2235 睹
2236 赌	2237 杜	2238 镀	2239 肚	2240 度

2241 渡　2242 妒　2243 端　2244 短　2245 锻
2246 段　2247 断　2248 缎　2249 堆　2250 兑
2251 队　2252 对　2253 墩　2254 吨　2255 蹲
2256 敦　2257 顿　2258 囤　2259 钝　2260 盾
2261 遁　2262 掇　2263 哆　2264 多　2265 夺
2266 垛　2267 躲　2268 朵　2269 跺　2270 舵
2271 剁　2272 惰　2273 堕　2274 蛾　2275 峨
2276 鹅　2277 俄　2278 额　2279 讹　2280 娥
2281 恶　2282 厄　2283 扼　2284 遏　2285 鄂
2286 饿　2287 恩　2288 而　2289 儿　2290 耳
2291 尔　2292 饵　2293 洱　2294 二

第 23 区

2301 贰　2302 发　2303 罚　2304 筏　2305 伐
2306 乏　2307 阀　2308 法　2309 珐　2310 藩
2311 帆　2312 番　2313 翻　2314 樊　2315 矾
2316 钒　2317 繁　2318 凡　2319 烦　2320 反
2321 返　2322 范　2323 贩　2324 犯　2325 饭
2326 泛　2327 坊　2328 芳　2329 方　2330 肪
2331 房　2332 防　2333 妨　2334 仿　2335 访
2336 纺　2337 放　2338 菲　2339 非　2340 啡
2341 飞　2342 肥　2343 匪　2344 诽　2345 吠
2346 肺　2347 废　2348 沸　2349 费　2350 芬
2351 酚　2352 吩　2353 氛　2354 分　2355 纷
2356 坟　2357 焚　2358 汾　2359 粉　2360 奋
2361 份　2362 忿　2363 愤　2364 粪　2365 丰
2366 封　2367 枫　2368 蜂　2369 峰　2370 锋
2371 风　2372 疯　2373 烽　2374 逢　2375 冯

2376 缝 2377 讽 2378 奉 2379 凤 2380 佛
2381 否 2382 夫 2383 敷 2384 肤 2385 孵
2386 扶 2387 拂 2388 辐 2389 幅 2390 氟
2391 符 2392 伏 2393 俘 2394 服

第 24 区

2401 浮 2402 涪 2403 福 2404 袱 2405 弗
2406 甫 2407 抚 2408 辅 2409 俯 2410 釜
2411 斧 2412 脯 2413 腑 2414 府 2415 腐
2416 赴 2417 副 2418 覆 2419 赋 2420 复
2421 傅 2422 付 2423 阜 2424 父 2425 腹
2426 负 2427 富 2428 讣 2429 附 2430 妇
2431 缚 2432 咐 2433 噶 2434 嘎 2435 该
2436 改 2437 概 2438 钙 2439 盖 2440 溉
2441 干 2442 甘 2443 杆 2444 柑 2445 竿
2446 肝 2447 赶 2448 感 2449 秆 2450 敢
2451 赣 2452 冈 2453 刚 2454 钢 2455 缸
2456 肛 2457 纲 2458 岗 2459 港 2460 杠
2461 篙 2462 皋 2463 高 2464 膏 2465 羔
2466 糕 2467 搞 2468 镐 2469 稿 2470 告
2471 哥 2472 歌 2473 搁 2474 戈 2475 鸽
2476 胳 2477 疙 2478 割 2479 革 2480 葛
2481 格 2482 蛤 2483 阁 2484 隔 2485 铬
2486 个 2487 各 2488 给 2489 根 2490 跟
2491 耕 2492 更 2493 庚 2494 羹

第 25 区

2501 埂 2502 耿 2503 梗 2504 工 2505 攻
2506 功 2507 恭 2508 龚 2509 供 2510 躬
2511 公 2512 宫 2513 弓 2514 巩 2515 汞
2516 拱 2517 贡 2518 共 2519 钩 2520 勾
2521 沟 2522 苟 2523 狗 2524 垢 2525 构
2526 购 2527 够 2528 辜 2529 菇 2530 咕
2531 箍 2532 估 2533 沽 2534 孤 2535 姑
2536 鼓 2537 古 2538 蛊 2539 骨 2540 谷
2541 股 2542 故 2543 顾 2544 固 2545 雇
2546 刮 2547 瓜 2548 剐 2549 寡 2550 挂
2551 褂 2552 乖 2553 拐 2554 怪 2555 棺
2556 关 2557 官 2558 冠 2559 观 2560 管
2561 馆 2562 罐 2563 惯 2564 灌 2565 贯
2566 光 2567 广 2568 逛 2569 瑰 2570 规
2571 圭 2572 硅 2573 归 2574 龟 2575 闺
2576 轨 2577 鬼 2578 诡 2579 癸 2580 桂
2581 柜 2582 跪 2583 贵 2584 刽 2585 辊
2586 滚 2587 棍 2588 锅 2589 郭 2590 国
2591 果 2592 裹 2593 过 2594 哈

第 26 区

2601 骸 2602 孩 2603 海 2604 氦 2605 亥
2606 害 2607 骇 2608 酣 2609 憨 2610 邯
2611 韩 2612 含 2613 涵 2614 寒 2615 函
2616 喊 2617 罕 2618 翰 2619 撼 2620 捍
2621 旱 2622 憾 2623 悍 2624 焊 2625 汗

2626 汉　2627 夯　2628 杭　2629 航　2630 壕
2631 嚎　2632 豪　2633 毫　2634 郝　2635 好
2636 耗　2637 号　2638 浩　2639 呵　2640 喝
2641 荷　2642 菏　2643 核　2644 禾　2645 和
2646 何　2647 合　2648 盒　2649 貉　2650 阂
2651 河　2652 涸　2653 赫　2654 褐　2655 鹤
2656 贺　2657 嘿　2658 黑　2659 痕　2660 很
2661 狠　2662 恨　2663 哼　2664 亨　2665 横
2666 衡　2667 恒　2668 轰　2669 哄　2670 烘
2671 虹　2672 鸿　2673 洪　2674 宏　2675 弘
2576 红　2677 喉　2678 侯　2679 猴　2680 吼
2581 厚　2682 候　2683 后　2684 呼　2685 乎
2686 忽　2687 瑚　2688 壶　2689 葫　2690 胡
2691 蝴　2692 狐　2693 糊　2694 湖

第 27 区

2701 弧　2702 虎　2703 唬　2704 护　2705 互
2706 沪　2707 户　2708 花　2709 哗　2710 华
2711 猾　2712 滑　2713 画　2714 划　2715 化
2716 话　2717 槐　2718 徊　2719 怀　2720 淮
2721 坏　2722 欢　2723 环　2724 桓　2725 还
2726 缓　2727 换　2728 患　2729 唤　2730 痪
2731 豢　2732 焕　2733 涣　2734 宦　2735 幻
2736 荒　2737 慌　2738 黄　2739 磺　2740 蝗
2741 簧　2742 皇　2743 凰　2744 惶　2745 煌
2746 晃　2747 幌　2748 恍　2749 谎　2750 灰
2751 挥　2752 辉　2753 徽　2754 恢　2755 蛔
2756 回　2757 毁　2758 悔　2759 慧　2760 卉

2761 惠 2762 晦 2763 贿 2764 秽 2765 会
2766 烩 2767 汇 2768 讳 2769 诲 2770 绘
2771 荤 2772 昏 2773 婚 2774 魂 2775 浑
2776 混 2777 豁 2778 活 2779 伙 2780 火
2781 获 2782 或 2783 惑 2784 霍 2785 货
2786 祸 2787 击 2788 圾 2789 基 2790 机
2791 畸 2792 稽 2793 积 2794 箕

第 28 区

2801 肌 2802 饥 2803 迹 2804 激 2805 讥
2806 鸡 2807 姬 2808 绩 2809 缉 2810 吉
2811 极 2812 棘 2813 辑 2814 籍 2815 集
2816 及 2817 急 2818 疾 2819 汲 2820 即
2821 嫉 2822 级 2823 挤 2824 几 2825 脊
2826 己 2827 蓟 2828 技 2829 冀 2830 季
2831 伎 2832 祭 2833 剂 2834 悸 2835 济
2836 寄 2837 寂 2838 计 2839 记 2840 既
2841 忌 2842 际 2843 妓 2844 继 2845 纪
2846 嘉 2847 枷 2848 夹 2849 佳 2850 家
2851 加 2852 荚 2853 颊 2854 贾 2855 甲
2856 钾 2857 假 2858 稼 2859 价 2860 架
2861 驾 2862 嫁 2863 歼 2864 监 2865 坚
2866 尖 2867 笺 2868 间 2869 煎 2870 兼
2871 肩 2872 艰 2873 奸 2874 缄 2875 茧
2876 检 2877 柬 2878 碱 2879 硷 2880 拣
2881 捡 2882 简 2883 俭 2884 剪 2885 减
2886 荐 2887 槛 2888 鉴 2889 践 2890 贱
2891 见 2892 键 2893 箭 2894 件

第 29 区

2901 健	2902 舰	2903 剑	2904 饯	2905 渐
2906 溅	2907 涧	2908 建	2909 僵	2910 姜
2911 将	2912 浆	2913 江	2914 疆	2915 蒋
2916 桨	2917 奖	2918 讲	2919 匠	2920 酱
2921 降	2922 蕉	2923 椒	2924 礁	2925 焦
2926 胶	2927 交	2928 郊	2929 浇	2930 骄
2931 娇	2932 嚼	2933 搅	2934 铰	2935 矫
2936 侥	2937 脚	2938 狡	2939 角	2940 饺
2941 缴	2942 绞	2943 剿	2944 教	2945 酵
2946 轿	2947 较	2948 叫	2949 窖	2950 揭
2951 接	2952 皆	2953 秸	2954 街	2955 阶
2956 截	2957 劫	2958 节	2959 桔	2960 杰
2961 捷	2962 睫	2963 竭	2964 洁	2965 结
2966 解	2967 姐	2968 戒	2969 藉	2970 芥
2971 界	2972 借	2973 介	2974 疥	2975 诫
2976 届	2977 巾	2978 筋	2979 斤	2980 金
2981 今	2982 津	2983 襟	2984 紧	2985 锦
2986 仅	2987 谨	2988 进	2989 靳	2990 晋
2991 禁	2992 近	2993 烬	2994 浸	

第 30 区

3001 尽	3002 劲	3003 荆	3004 兢	3005 茎
3006 睛	3007 晶	3008 鲸	3009 京	3010 惊
3011 精	3012 粳	3013 经	3014 井	3015 警
3016 景	3017 颈	3018 静	3019 境	3020 敬
3021 镜	3022 径	3023 痉	3024 靖	3025 竟

3026 竞	3027 净	3028 炯	3029 窘	3030 揪
3031 究	3032 纠	3033 玖	3034 韭	3035 久
3036 灸	3037 九	3038 酒	3039 厩	3040 救
3041 旧	3042 臼	3043 舅	3044 咎	3045 就
3046 疚	3047 鞠	3048 拘	3049 狙	3050 疽
3051 居	3052 驹	3053 菊	3054 局	3055 咀
3056 矩	3057 举	3058 沮	3059 聚	3060 拒
3061 据	3062 巨	3063 具	3064 距	3065 踞
3066 锯	3067 俱	3068 句	3069 惧	3070 炬
3071 剧	3072 捐	3073 鹃	3074 娟	3075 倦
3076 眷	3077 卷	3078 绢	3079 撅	3080 攫
3081 抉	3082 掘	3083 倔	3084 爵	3085 觉
3086 决	3087 诀	3088 绝	3089 均	3090 菌
3091 钧	3092 军	3093 君	3094 峻	

第 31 区

3101 俊	3102 竣	3103 浚	3104 郡	3105 骏
3106 喀	3107 咖	3108 卡	3109 咯	3110 开
3111 揩	3112 楷	3113 凯	3114 慨	3115 刊
3116 堪	3117 勘	3118 坎	3119 砍	3120 看
3121 康	3122 慷	3123 糠	3124 扛	3125 抗
3126 亢	3127 炕	3128 考	3129 拷	3130 烤
3131 靠	3132 坷	3133 苛	3134 柯	3135 棵
3136 磕	3137 颗	3138 科	3139 壳	3140 咳
3141 可	3142 渴	3143 克	3144 刻	3145 客
3146 课	3147 肯	3148 啃	3149 垦	3150 恳
3151 坑	3152 吭	3153 空	3154 恐	3155 孔
3156 控	3157 抠	3158 口	3159 扣	3160 寇

3161 枯 3162 哭 3163 窟 3164 苦 3165 酷
3166 库 3167 裤 3168 夸 3169 垮 3170 挎
3171 跨 3172 胯 3173 块 3174 筷 3175 侩
3176 快 3177 宽 3178 款 3179 匡 3180 筐
3181 狂 3182 框 3183 矿 3184 眶 3185 旷
3186 况 3187 亏 3188 盔 3189 岿 3190 窥
3191 葵 3192 奎 3193 魁 3194 傀

第 32 区

3201 馈 3202 愧 3203 溃 3204 坤 3205 昆
3206 捆 3207 困 3208 括 3209 扩 3210 廓
3211 阔 3212 垃 3213 拉 3214 喇 3215 蜡
3216 腊 3217 辣 3218 啦 3219 莱 3220 来
3221 赖 3222 蓝 3223 婪 3224 栏 3225 拦
3226 篮 3227 阑 3228 兰 3229 澜 3230 谰
3231 揽 3232 览 3233 懒 3234 缆 3235 烂
3236 滥 3237 琅 3238 榔 3239 狼 3240 廊
3241 郎 3242 朗 3243 浪 3244 捞 3245 劳
3246 牢 3247 老 3248 佬 3249 姥 3250 酪
3251 烙 3252 涝 3253 勒 3254 乐 3255 雷
3256 镭 3257 蕾 3258 磊 3259 累 3260 儡
3261 垒 3262 擂 3263 肋 3264 类 3265 泪
3266 棱 3267 楞 3268 冷 3269 厘 3270 梨
3271 犁 3272 黎 3273 篱 3274 狸 3275 离
3276 漓 3277 理 3278 李 3279 里 3280 鲤
3281 礼 3282 莉 3283 荔 3284 吏 3285 栗
3286 丽 3287 厉 3288 励 3289 砾 3290 历
3291 利 3292 傈 3293 例 3294 俐

第 33 区

3301 痢 3302 立 3303 粒 3304 沥 3305 隶
3306 力 3307 璃 3308 哩 3309 俩 3310 联
3311 莲 3312 连 3313 镰 3314 廉 3315 怜
3316 涟 3317 帘 3318 敛 3319 脸 3320 链
3321 恋 3322 炼 3323 练 3324 粮 3325 凉
3326 梁 3327 粱 3328 良 3329 两 3330 辆
3331 量 3332 晾 3333 亮 3334 谅 3335 撩
3336 聊 3337 僚 3338 疗 3339 燎 3340 寥
3341 辽 3342 潦 3343 了 3344 撂 3345 镣
3346 廖 3347 料 3348 列 3349 裂 3350 烈
3351 劣 3352 猎 3353 琳 3354 林 3355 磷
3356 霖 3357 临 3358 邻 3359 鳞 3360 淋
3361 凛 3362 赁 3363 吝 3364 拎 3365 玲
3366 菱 3367 零 3368 龄 3369 铃 3370 伶
3371 羚 3372 凌 3373 灵 3374 陵 3375 岭
3376 领 3377 另 3378 令 3379 溜 3380 琉
3381 榴 3382 硫 3383 馏 3384 留 3385 刘
3386 瘤 3387 流 3388 柳 3389 六 3390 龙
3391 聋 3392 咙 3393 笼 3394 窿

第 34 区

3401 隆 3402 垄 3403 拢 3404 陇 3405 楼
3406 娄 3407 搂 3408 篓 3409 漏 3410 陋
3411 芦 3412 卢 3413 颅 3414 庐 3415 炉
3416 掳 3417 卤 3418 虏 3419 鲁 3420 麓
3421 碌 3422 露 3423 路 3424 赂 3425 鹿

3426 潞 3427 禄 3428 录 3429 陆 3430 戮
3431 驴 3432 吕 3433 铝 3434 侣 3435 旅
3436 履 3437 屡 3438 缕 3439 虑 3440 氯
3441 律 3442 率 3443 滤 3444 绿 3445 峦
3446 挛 3447 孪 3448 滦 3449 卵 3450 乱
3451 掠 3452 略 3453 抡 3454 轮 3455 伦
3456 仑 3457 沦 3458 纶 3459 论 3460 萝
3461 螺 3462 罗 3463 逻 3464 锣 3465 箩
3466 骡 3467 裸 3468 落 3469 洛 3470 骆
3471 络 3472 妈 3473 麻 3474 玛 3475 码
3476 蚂 3477 马 3478 骂 3479 嘛 3480 吗
3481 埋 3482 买 3483 麦 3484 卖 3485 迈
3486 脉 3487 瞒 3488 馒 3489 蛮 3490 满
3491 蔓 3492 曼 3493 慢 3494 漫

第 35 区

3501 谩 3502 芒 3503 茫 3504 盲 3505 氓
3506 忙 3507 莽 3508 猫 3509 茅 3510 锚
3511 毛 3512 矛 3513 铆 3514 卯 3515 茂
3516 冒 3517 帽 3518 貌 3519 贸 3520 么
3521 玫 3522 枚 3523 梅 3524 酶 3525 霉
3526 煤 3527 没 3528 眉 3529 媒 3530 镁
3531 每 3532 美 3533 昧 3534 寐 3535 妹
3536 媚 3537 门 3538 闷 3539 们 3540 萌
3541 蒙 3542 檬 3543 盟 3544 锰 3545 猛
3546 梦 3547 孟 3548 眯 3549 醚 3550 靡
3551 糜 3552 迷 3553 谜 3554 弥 3555 米
3556 秘 3557 觅 3558 泌 3559 蜜 3560 密

3561 幂 3562 棉 3563 眠 3564 绵 3565 冕
3566 免 3567 勉 3568 娩 3569 缅 3570 面
3571 苗 3572 描 3573 瞄 3574 藐 3575 秒
3576 渺 3577 庙 3578 妙 3579 蔑 3580 灭
3581 民 3582 抿 3583 皿 3584 敏 3585 悯
3586 闽 3587 明 3588 螟 3589 鸣 3590 铭
3591 名 3592 命 3593 谬 3594 摸

第 36 区

3601 摹 3602 蘑 3603 模 3604 膜 3605 磨
3606 摩 3607 魔 3608 抹 3609 末 3610 莫
3611 墨 3612 默 3613 沫 3614 漠 3615 寞
3616 陌 3617 谋 3618 牟 3619 某 3620 拇
3621 牡 3622 亩 3623 姆 3624 母 3625 墓
3626 暮 3627 幕 3628 募 3629 慕 3630 木
3631 目 3632 睦 3633 牧 3634 穆 3635 拿
3636 哪 3637 呐 3638 钠 3639 那 3640 娜
3641 纳 3642 氖 3643 乃 3644 奶 3645 耐
3646 奈 3647 南 3648 男 3649 难 3650 囊
3651 挠 3652 脑 3653 恼 3654 闹 3655 淖
3656 呢 3657 馁 3658 内 3659 嫩 3660 能
3661 妮 3662 霓 3663 倪 3664 泥 3665 尼
3666 拟 3667 你 3668 匿 3669 腻 3670 逆
3671 溺 3672 蔫 3673 拈 3674 年 3675 碾
3676 撵 3677 捻 3678 念 3679 娘 3680 酿
3681 鸟 3682 尿 3683 捏 3684 聂 3685 孽
3686 啮 3687 镊 3688 镍 3689 涅 3690 您
3691 柠 3692 狞 3693 凝 3694 宁

第 37 区

3701 拧 3702 泞 3703 牛 3704 扭 3705 钮
3706 纽 3707 脓 3708 浓 3709 农 3710 弄
3711 奴 3712 努 3713 怒 3714 女 3715 暖
3716 虐 3717 疟 3718 挪 3719 懦 3720 糯
3721 诺 3722 哦 3723 欧 3724 鸥 3725 殴
3726 藕 3727 呕 3728 偶 3729 沤 3730 啪
3731 趴 3732 爬 3733 帕 3734 怕 3735 琶
3736 拍 3737 排 3738 牌 3739 徘 3740 湃
3741 派 3742 攀 3743 潘 3744 盘 3745 磐
3746 盼 3747 畔 3748 判 3749 叛 3750 乓
3751 庞 3752 旁 3753 耪 3754 胖 3755 抛
3756 咆 3757 刨 3758 炮 3759 袍 3760 跑
3761 泡 3762 呸 3763 胚 3764 培 3765 裴
3766 赔 3767 陪 3768 配 3769 佩 3770 沛
3771 喷 3772 盆 3773 砰 3774 抨 3775 烹
3776 澎 3777 彭 3778 蓬 3779 棚 3780 硼
3781 篷 3782 膨 3783 朋 3784 鹏 3785 捧
3786 碰 3787 坯 3788 砒 3789 霹 3790 批
3791 披 3792 劈 3793 琵 3794 毗

第 38 区

3801 啤 3802 脾 3803 疲 3804 皮 3805 匹
3806 痞 3807 僻 3808 屁 3809 譬 3810 篇
3811 偏 3812 片 3813 骗 3814 飘 3815 漂
3816 瓢 3817 票 3818 撇 3819 瞥 3820 拼
3821 频 3822 贫 3823 品 3824 聘 3825 乒

3826 坪　3827 苹　3828 萍　3829 平　3830 凭
3831 瓶　3832 评　3833 屏　3834 坡　3835 泼
3836 颇　3837 婆　3838 破　3839 魄　3840 迫
3841 粕　3842 剖　3843 扑　3844 铺　3845 仆
3846 莆　3847 葡　3848 菩　3849 蒲　3850 埔
3851 朴　3852 圃　3853 普　3854 浦　3855 谱
3856 曝　3857 瀑　3858 期　3859 欺　3860 栖
3861 戚　3862 妻　3863 七　3864 凄　3865 漆
3866 柒　3867 沏　3868 其　3869 棋　3870 奇
3871 歧　3872 畦　3873 崎　3874 脐　3875 齐
3876 旗　3877 祈　3878 祁　3879 骑　3880 起
3881 岂　3882 乞　3883 企　3884 启　3885 契
3886 砌　3887 器　3888 气　3889 迄　3890 弃
3891 汽　3892 泣　3893 讫　3894 掐

第 39 区

3901 恰　3902 洽　3903 牵　3904 扦　3905 钎
3906 铅　3907 千　3908 迁　3909 签　3910 仟
3911 谦　3912 乾　3913 黔　3914 钱　3915 钳
3916 前　3917 潜　3918 遣　3919 浅　3920 谴
3921 堑　3922 嵌　3923 欠　3924 歉　3925 枪
3926 呛　3927 腔　3928 羌　3929 墙　3930 蔷
3931 强　3932 抢　3933 橇　3934 锹　3935 敲
3936 悄　3937 桥　3938 瞧　3939 乔　3940 侨
3941 巧　3942 鞘　3943 撬　3944 翘　3945 峭
3946 俏　3947 窍　3948 切　3949 茄　3950 且

3951 怯　3952 窃　3953 钦　3954 侵　3955 亲
3956 秦　3957 琴　3958 勤　3959 芹　3960 擒
3961 禽　3962 寝　3963 沁　3964 青　3965 轻
3966 氢　3967 倾　3968 卿　3969 清　3970 擎
3971 晴　3972 氰　3973 情　3974 顷　3975 请
3976 庆　3977 琼　3978 穷　3979 秋　3980 丘
3981 邱　3982 球　3983 求　3984 囚　3985 酋
3986 泅　3987 趋　3988 区　3989 蛆　3990 曲
3991 躯　3992 屈　3993 驱　3994 渠

第 40 区

4001 取　4002 娶　4003 龋　4004 趣　4005 去
4006 圈　4007 颧　4008 权　4009 醛　4010 泉
4011 全　4012 痊　4013 拳　4014 犬　4015 券
4016 劝　4017 缺　4018 炔　4019 瘸　4020 却
4021 鹊　4022 榷　4023 确　4024 雀　4025 裙
4026 群　4027 然　4028 燃　4029 冉　4030 染
4031 瓤　4032 壤　4033 攘　4034 嚷　4035 让
4036 饶　4037 扰　4038 绕　4039 惹　4040 热
4041 壬　4042 仁　4043 人　4044 忍　4045 韧
4046 任　4047 认　4048 刃　4049 妊　4050 纫
4051 扔　4052 仍　4053 日　4054 戎　4055 茸
4056 蓉　4057 荣　4058 融　4059 熔　4060 溶
4061 容　4062 绒　4063 冗　4064 揉　4065 柔
4066 肉　4067 茹　4068 蠕　4069 儒　4070 孺
4071 如　4072 辱　4073 乳　4074 汝　4075 入

4076 褥 4077 软 4078 阮 4079 蕊 4080 瑞
4081 锐 4082 闰 4083 润 4084 若 4085 弱
4086 撒 4087 洒 4088 萨 4089 腮 4090 鳃
4091 塞 4092 赛 4093 三 4094 叁

第 41 区

4101 伞 4102 散 4103 桑 4104 嗓 4105 丧
4106 搔 4107 骚 4108 扫 4109 嫂 4110 瑟
4111 色 4112 涩 4113 森 4114 僧 4115 莎
4116 砂 4117 杀 4118 刹 4119 沙 4120 纱
4121 傻 4122 啥 4123 煞 4124 筛 4125 晒
4126 珊 4127 苫 4128 杉 4129 山 4130 删
4131 煽 4132 衫 4133 闪 4134 陕 4135 擅
4136 赡 4137 膳 4138 善 4139 汕 4140 扇
4141 缮 4142 墒 4143 伤 4144 商 4145 赏
4146 晌 4147 上 4148 尚 4149 裳 4150 梢
4151 捎 4152 稍 4153 烧 4154 芍 4155 勺
4156 韶 4157 少 4158 哨 4159 邵 4160 绍
4161 奢 4162 赊 4163 蛇 4164 舌 4165 舍
4166 赦 4167 摄 4168 射 4169 慑 4170 涉
4171 社 4172 设 4173 砷 4174 申 4175 呻
4176 伸 4177 身 4178 深 4179 娠 4180 绅
4181 神 4182 沈 4183 审 4184 婶 4185 甚
4186 肾 4187 慎 4188 渗 4189 声 4190 生
4191 甥 4192 牲 4193 升 4194 绳

第 42 区

4201 省　4202 盛　4203 剩　4204 胜　4205 圣
4206 师　4207 失　4208 狮　4209 施　4210 湿
4211 诗　4212 尸　4213 虱　4214 十　4215 石
4216 拾　4217 时　4218 什　4219 食　4220 蚀
4221 实　4222 识　4223 史　4224 矢　4225 使
4226 屎　4227 驶　4228 始　4229 式　4230 示
4231 士　4232 世　4233 柿　4234 事　4235 拭
4236 誓　4237 逝　4238 势　4239 是　4240 嗜
4241 噬　4242 适　4243 仕　4244 侍　4245 释
4246 饰　4247 氏　4248 市　4249 恃　4250 室
4251 视　4252 试　4253 收　4254 手　4255 首
4256 守　4257 寿　4258 授　4259 售　4260 受
4261 瘦　4262 兽　4263 蔬　4264 枢　4265 梳
4266 殊　4267 抒　4268 输　4269 叔　4270 舒
4271 淑　4272 疏　4273 书　4274 赎　4275 孰
4276 熟　4277 薯　4278 暑　4279 曙　4280 署
4281 蜀　4282 黍　4283 鼠　4284 属　4285 术
4286 述　4287 树　4288 束　4289 戍　4290 竖
4291 墅　4292 庶　4293 数　4294 漱

第 43 区

4301 恕　4302 刷　4303 耍　4304 摔　4305 衰
4306 甩　4307 帅　4308 栓　4309 拴　4310 霜
4311 双　4312 爽　4313 谁　4314 水　4315 睡
4316 税　4317 吮　4318 瞬　4319 顺　4320 舜

4321 说 4322 硕 4323 朔 4324 烁 4325 斯
4326 撕 4327 嘶 4328 思 4329 私 4330 司
4331 丝 4332 死 4333 肆 4334 寺 4335 嗣
4336 四 4337 伺 4338 似 4339 饲 4340 巳
4341 松 4342 耸 4343 怂 4344 颂 4345 送
4346 宋 4347 讼 4348 诵 4349 搜 4350 艘
4351 擞 4352 嗽 4353 苏 4354 酥 4355 俗
4356 素 4357 速 4358 粟 4359 僳 4360 塑
4361 溯 4362 宿 4363 诉 4364 肃 4365 酸
4366 蒜 4367 算 4368 虽 4369 隋 4370 随
4371 绥 4372 髓 4373 碎 4374 岁 4375 穗
4376 遂 4377 隧 4378 祟 4379 孙 4380 损
4381 笋 4382 蓑 4383 梭 4384 唆 4385 缩
4386 琐 4387 索 4388 锁 4389 所 4390 塌
4391 他 4392 它 4393 她 4394 塔

第 44 区

4401 獭 4402 挞 4403 蹋 4404 踏 4405 胎
4406 苔 4407 抬 4408 台 4409 泰 4410 酞
4411 太 4412 态 4413 汰 4414 坍 4415 摊
4416 贪 4417 瘫 4418 滩 4419 坛 4420 檀
4421 痰 4422 潭 4423 谭 4424 谈 4425 坦
4426 毯 4427 袒 4428 碳 4429 探 4430 叹
4431 炭 4432 汤 4433 塘 4434 搪 4435 堂
4436 棠 4437 膛 4438 唐 4439 糖 4440 倘
4441 躺 4442 淌 4443 趟 4444 烫 4445 掏
4446 涛 4447 滔 4448 绦 4449 萄 4450 桃

4451 逃　4452 淘　4453 陶　4454 讨　4455 套
4456 特　4457 藤　4458 腾　4459 疼　4460 誊
4461 梯　4462 剔　4463 踢　4464 锑　4465 提
4466 题　4467 蹄　4468 啼　4469 体　4470 替
4471 嚏　4472 惕　4473 涕　4474 剃　4475 屉
4476 天　4477 添　4478 填　4479 田　4480 甜
4481 恬　4482 舔　4483 腆　4484 挑　4485 条
4486 迢　4487 眺　4488 跳　4489 贴　4490 铁
4491 帖　4492 厅　4493 听　4494 烃

第 45 区

4501 汀　4502 廷　4503 停　4504 亭　4505 庭
4506 挺　4507 艇　4508 通　4509 桐　4510 酮
4511 瞳　4512 同　4513 铜　4514 彤　4515 童
4516 桶　4517 捅　4518 筒　4519 统　4520 痛
4521 偷　4522 投　4523 头　4524 透　4525 凸
4526 秃　4527 突　4528 图　4529 徒　4530 途
4531 涂　4532 屠　4533 土　4534 吐　4535 兔
4536 湍　4537 团　4538 推　4539 颓　4540 腿
4541 蜕　4542 褪　4543 退　4544 吞　4545 屯
4546 臀　4547 拖　4548 托　4549 脱　4550 鸵
4551 陀　4552 驮　4553 驼　4554 椭　4555 妥
4556 拓　4557 唾　4558 挖　4559 哇　4560 蛙
4561 洼　4562 娃　4563 瓦　4564 袜　4565 歪
4566 外　4567 豌　4568 弯　4569 湾　4570 玩
4571 顽　4572 丸　4573 烷　4574 完　4575 碗
4576 挽　4577 晚　4578 皖　4579 惋　4580 宛

4581 婉 4582 万 4583 腕 4584 汪 4585 王
4586 亡 4587 枉 4588 网 4589 往 4590 旺
4591 望 4592 忘 4593 妄 4594 威

第 46 区

4601 巍 4602 微 4603 危 4604 韦 4605 违
4606 桅 4607 围 4608 唯 4609 惟 4610 为
4611 潍 4612 维 4613 苇 4614 萎 4615 委
4616 伟 4617 伪 4618 尾 4619 纬 4620 未
4621 蔚 4622 味 4623 畏 4624 胃 4625 喂
4626 魏 4627 位 4628 渭 4629 谓 4630 尉
4631 慰 4632 卫 4633 瘟 4634 温 4635 蚊
4636 文 4637 闻 4638 纹 4639 吻 4640 稳
4641 紊 4642 问 4643 嗡 4644 翁 4645 瓮
4646 挝 4647 蜗 4648 涡 4649 窝 4650 我
4651 斡 4652 卧 4653 握 4654 沃 4655 巫
4656 呜 4657 钨 4658 乌 4659 污 4660 诬
4661 屋 4662 无 4663 芜 4664 梧 4665 吾
4666 吴 4667 毋 4668 武 4669 五 4670 捂
4671 午 4672 舞 4673 伍 4674 侮 4675 坞
4676 戊 4677 雾 4678 晤 4679 物 4680 勿
4681 务 4682 悟 4683 误 4684 昔 4685 熙
4686 析 4687 西 4688 硒 4689 矽 4690 晰
4691 嘻 4692 吸 4693 锡 4694 牺

第 47 区

4701 稀 4702 息 4703 希 4704 悉 4705 膝
4706 夕 4707 惜 4708 熄 4709 烯 4710 溪
4711 汐 4712 犀 4713 檄 4714 袭 4715 席
4716 习 4717 媳 4718 喜 4719 铣 4720 洗
4721 系 4722 隙 4723 戏 4724 细 4725 瞎
4726 虾 4727 匣 4728 霞 4729 辖 4730 暇
4731 峡 4732 侠 4733 狭 4734 下 4735 厦
4736 夏 4737 吓 4738 掀 4739 锨 4740 先
4741 仙 4742 鲜 4743 纤 4744 咸 4745 贤
4746 衔 4747 舷 4748 闲 4749 涎 4750 弦
4751 嫌 4752 显 4753 险 4754 现 4755 献
4756 县 4757 腺 4758 馅 4759 羡 4760 宪
4761 陷 4762 限 4763 线 4764 相 4765 厢
4766 镶 4767 香 4768 箱 4769 襄 4770 湘
4771 乡 4772 翔 4773 祥 4774 详 4775 想
4776 响 4777 享 4778 项 4779 巷 4780 橡
4781 像 4782 向 4783 象 4784 萧 4785 硝
4786 霄 4787 削 4788 哮 4789 嚣 4790 销
4791 消 4792 宵 4793 淆 4794 晓

第 48 区

4801 小 4802 孝 4803 校 4804 肖 4805 啸
4806 笑 4807 效 4808 楔 4809 些 4810 歇
4811 蝎 4812 鞋 4813 协 4814 挟 4815 携
4816 邪 4817 斜 4818 胁 4819 谐 4820 写

4821 械　4822 卸　4823 蟹　4824 懈　4825 泄
4826 泻　4827 谢　4828 屑　4829 薪　4830 芯
4831 锌　4832 欣　4833 辛　4834 新　4835 忻
4836 心　4837 信　4838 衅　4839 星　4840 腥
4841 猩　4842 惺　4843 兴　4844 刑　4845 型
4846 形　4847 邢　4848 行　4849 醒　4850 幸
4851 杏　4852 性　4853 姓　4854 兄　4855 凶
4856 胸　4857 匈　4858 汹　4859 雄　4860 熊
4861 休　4862 修　4863 羞　4864 朽　4865 嗅
4866 锈　4867 秀　4868 袖　4869 绣　4870 墟
4871 戌　4872 需　4873 虚　4874 嘘　4875 须
4876 徐　4877 许　4878 蓄　4879 酗　4880 叙
4881 旭　4882 序　4883 畜　4884 恤　4885 絮
4886 婿　4887 绪　4888 续　4889 轩　4890 喧
4891 宣　4892 悬　4893 旋　4894 玄

第 49 区

4901 选　4902 癣　4903 眩　4904 绚　4905 靴
4906 薛　4907 学　4908 穴　4909 雪　4910 血
4911 勋　4912 熏　4913 循　4914 旬　4915 询
4916 寻　4917 驯　4918 巡　4919 殉　4920 汛
4921 训　4922 讯　4923 逊　4924 迅　4925 压
4926 押　4927 鸦　4928 鸭　4929 呀　4930 丫
4931 芽　4932 牙　4933 蚜　4934 崖　4935 衙
4936 涯　4937 雅　4938 哑　4939 亚　4940 讶
4941 焉　4942 咽　4943 阉　4944 烟　4945 淹
4946 盐　4947 严　4948 研　4949 蜒　4950 岩

4951 延 4952 言 4953 颜 4954 阎 4955 炎
4956 沿 4957 奄 4958 掩 4959 眼 4960 衍
4961 演 4962 艳 4963 堰 4964 燕 4965 厌
4966 砚 4967 雁 4968 唁 4969 彦 4970 焰
4971 宴 4972 谚 4973 验 4974 殃 4975 央
4976 鸯 4977 秧 4978 杨 4979 扬 4980 佯
4981 疡 4982 羊 4983 洋 4984 阳 4985 氧
4986 仰 4987 痒 4988 养 4989 样 4990 漾
4991 邀 4992 腰 4993 妖 4994 瑶

第 50 区

5001 摇 5002 尧 5003 遥 5004 窑 5005 谣
5006 姚 5007 咬 5008 舀 5009 药 5010 要
5011 耀 5012 椰 5013 噎 5014 耶 5015 爷
5016 野 5017 冶 5018 也 5019 页 5020 掖
5021 业 5022 叶 5023 曳 5024 腋 5025 夜
5026 液 5027 一 5028 壹 5029 医 5030 揖
5031 铱 5032 依 5033 伊 5034 衣 5035 颐
5036 夷 5037 遗 5038 移 5039 仪 5040 胰
5041 疑 5042 沂 5043 宜 5044 姨 5045 彝
5046 椅 5047 蚁 5048 倚 5049 已 5050 乙
5051 矣 5052 以 5053 艺 5054 抑 5055 易
5056 邑 5057 屹 5058 亿 5059 役 5060 臆
5061 逸 5062 肄 5063 疫 5064 亦 5065 裔
5066 意 5067 毅 5068 忆 5069 义 5070 益
5071 溢 5072 诣 5073 议 5074 谊 5075 译
5076 异 5077 翼 5078 翌 5079 绎 5080 茵

5081 荫 5082 因 5083 殷 5084 音 5085 阴
5086 姻 5087 吟 5088 银 5089 淫 5090 寅
5091 饮 5092 尹 5093 引 5094 隐

第 51 区

5101 印 5102 英 5103 樱 5104 婴 5105 鹰
5106 应 5107 缨 5108 莹 5109 萤 5110 营
5111 荧 5112 蝇 5113 迎 5114 赢 5115 盈
5116 影 5117 颖 5118 硬 5119 映 5120 哟
5121 拥 5122 佣 5123 臃 5124 痈 5125 庸
5126 雍 5127 踊 5128 蛹 5129 咏 5130 泳
5131 涌 5132 永 5133 恿 5134 勇 5135 用
5136 幽 5137 优 5138 悠 5139 忧 5140 尤
5141 由 5142 邮 5143 铀 5144 犹 5145 油
5146 游 5147 酉 5148 有 5149 友 5150 右
5151 佑 5152 釉 5153 诱 5154 又 5155 幼
5156 迂 5157 淤 5158 于 5159 盂 5160 榆
5161 虞 5162 愚 5163 舆 5164 余 5165 俞
5166 逾 5167 鱼 5168 愉 5169 渝 5170 渔
5171 隅 5172 予 5173 娱 5174 雨 5175 与
5176 屿 5177 禹 5178 宇 5179 语 5180 羽
5181 玉 5182 域 5183 芋 5184 郁 5185 吁
5186 遇 5187 喻 5188 峪 5189 御 5190 愈
5191 欲 5192 狱 5193 育 5194 誉

第 52 区

5201 浴	5202 寓	5203 裕	5204 预	5205 豫
5206 驭	5207 鸳	5208 渊	5209 冤	5210 元
5211 垣	5212 袁	5213 原	5214 援	5215 辕
5216 园	5217 员	5218 圆	5219 猿	5220 源
5221 缘	5222 远	5223 苑	5224 愿	5225 怨
5226 院	5227 曰	5228 约	5229 越	5230 跃
5231 钥	5232 岳	5233 粤	5234 月	5235 悦
5236 阅	5237 耘	5238 云	5239 郧	5240 匀
5241 陨	5242 允	5243 运	5244 蕴	5245 酝
5246 晕	5247 韵	5248 孕	5249 匝	5250 砸
5251 杂	5252 栽	5253 哉	5254 灾	5255 宰
5256 载	5257 再	5258 在	5259 咱	5260 攒
5261 暂	5262 赞	5263 赃	5264 脏	5265 葬
5266 遭	5267 糟	5268 凿	5269 藻	5270 枣
5271 早	5272 澡	5273 蚤	5274 躁	5275 噪
5276 造	5277 皂	5278 灶	5279 燥	5280 责
5281 择	5282 则	5283 泽	5284 贼	5285 怎
5286 增	5287 憎	5288 曾	5289 赠	5290 扎
5291 喳	5292 渣	5293 札	5294 轧	

第 53 区

5301 铡	5302 闸	5303 眨	5304 栅	5305 榨
5306 咋	5307 乍	5308 炸	5309 诈	5310 摘
5311 斋	5312 宅	5313 窄	5314 债	5315 寨
5316 瞻	5317 毡	5318 詹	5319 粘	5320 沾

5321 盏	5322 斩	5323 辗	5324 崭	5325 展
5326 蘸	5327 栈	5328 占	5329 战	5330 站
5331 湛	5332 绽	5333 樟	5334 章	5335 彰
5336 漳	5337 张	5338 掌	5339 涨	5340 杖
5341 丈	5342 帐	5343 账	5344 仗	5345 胀
5346 瘴	5347 障	5348 招	5349 昭	5350 找
5351 沼	5352 赵	5353 照	5354 罩	5355 兆
5356 肇	5357 召	5358 遮	5359 折	5360 哲
5361 蛰	5362 辙	5363 者	5364 锗	5365 蔗
5366 这	5367 浙	5368 珍	5369 斟	5370 真
5371 甄	5372 砧	5373 臻	5374 贞	5375 针
5376 侦	5377 枕	5378 疹	5379 诊	5380 震
5381 振	5382 镇	5383 阵	5384 蒸	5385 挣
5386 睁	5387 征	5388 狰	5389 争	5390 怔
5391 整	5392 拯	5393 正	5394 政	

第 54 区

5401 帧	5402 症	5403 郑	5404 证	5405 芝
5406 枝	5407 支	5408 吱	5409 蜘	5410 知
5411 肢	5412 脂	5413 汁	5414 之	5415 织
5416 职	5417 直	5418 植	5419 殖	5420 执
5421 值	5422 侄	5423 址	5424 指	5425 止
5426 趾	5427 只	5428 旨	5429 纸	5430 志
5431 挚	5432 掷	5433 至	5434 致	5435 置
5436 帜	5437 峙	5438 制	5439 智	5440 秩
5441 稚	5442 质	5443 炙	5444 痔	5445 滞
5446 治	5447 窒	5448 中	5449 盅	5450 忠

5451 钟	5452 衷	5453 终	5454 种	5455 肿
5456 重	5457 仲	5458 众	5459 舟	5460 周
5461 州	5462 洲	5463 诌	5464 粥	5465 轴
5466 肘	5467 帚	5468 咒	5469 皱	5470 宙
5471 昼	5472 骤	5473 珠	5474 株	5475 蛛
5476 朱	5477 猪	5478 诸	5479 诛	5480 逐
5481 竹	5482 烛	5483 煮	5484 拄	5485 瞩
5486 嘱	5487 主	5488 著	5489 柱	5490 助
5491 蛀	5492 贮	5493 铸	5494 筑	

第 55 区

5501 住	5502 注	5503 祝	5504 驻	5505 抓
5506 爪	5507 拽	5508 专	5509 砖	5510 转
5511 撰	5512 赚	5513 篆	5514 桩	5515 庄
5516 装	5517 妆	5518 撞	5519 壮	5520 状
5521 椎	5522 锥	5523 追	5524 赘	5525 坠
5526 缀	5527 谆	5528 准	5529 捉	5530 拙
5531 卓	5532 桌	5533 琢	5534 茁	5535 酌
5536 啄	5537 着	5538 灼	5539 浊	5540 兹
5541 咨	5542 资	5543 姿	5544 滋	5545 淄
5546 孜	5547 紫	5548 仔	5549 籽	5550 滓
5551 子	5552 自	5553 渍	5554 字	5555 鬃
5556 棕	5557 踪	5558 宗	5559 综	5560 总
5561 纵	5562 邹	5563 走	5564 奏	5565 揍
5566 租	5567 足	5568 卒	5569 族	5570 祖
5571 诅	5572 阻	5573 组	5574 钻	5575 纂
5576 嘴	5577 醉	5578 最	5579 罪	5580 尊

5581 遵 5582 昨 5583 左 5584 佐 5585 柞
5586 做 5587 作 5588 坐 5589 座 5590
5591 5592 5593 5594

第 56 区

5601 亍 5602 丌 5603 兀 5604 丐 5605 廿
5606 卅 5607 丕 5608 亘 5609 丞 5610 鬲
5611 孬 5612 噩 5613 丨 5614 禺 5615 丿
5616 匕 5617 乇 5618 夭 5619 爻 5620 卮
5621 氐 5622 囟 5623 胤 5624 馗 5625 毓
5626 睾 5627 鼗 5628 丶 5629 亟 5630 鼐
5631 乜 5632 乩 5633 亓 5634 芈 5635 孛
5636 啬 5637 嘏 5638 仄 5639 厍 5640 厝
5641 厣 5642 厥 5643 厮 5644 靥 5645 赝
5646 匚 5647 叵 5648 匦 5649 匮 5650 匾
5651 赜 5652 卦 5653 卣 5654 刂 5655 刈
5656 刎 5657 刭 5658 刳 5659 刿 5660 剀
5661 剌 5662 剞 5663 剡 5664 剜 5665 蒯
5666 剽 5667 劂 5668 劁 5669 劐 5670 劓
5671 冂 5672 罔 5673 亻 5674 仃 5675 仉
5676 仂 5677 仨 5678 仡 5679 仫 5680 仞
5681 伛 5682 仳 5683 伢 5684 佤 5685 仵
5686 伥 5687 伧 5688 伉 5689 伫 5690 佞
5691 佧 5692 攸 5693 佚 5694 佝

第 57 区

5701 佟	5702 佗	5703 伲	5704 伽	5705 佶
5706 佴	5707 侑	5708 侉	5709 侃	5710 侏
5711 佾	5712 佻	5713 侪	5714 佼	5715 侬
5716 侔	5717 俦	5718 俨	5719 俪	5720 俅
5721 俚	5722 俣	5723 俜	5724 俑	5725 俟
5726 俸	5727 倩	5728 偌	5729 俳	5730 倬
5731 倏	5732 倮	5733 倭	5734 俾	5735 倜
5736 倌	5737 倥	5738 倨	5739 偾	5740 偃
5741 偕	5742 偈	5743 偎	5744 偬	5745 偻
5746 傥	5747 傧	5748 傩	5749 傺	5750 僖
5751 儆	5752 僭	5753 僬	5754 僦	5755 僮
5756 儇	5757 儋	5758 仝	5759 氽	5760 佘
5761 佥	5762 俎	5763 龠	5764 汆	5765 籴
5766 兮	5767 巽	5768 黉	5769 馘	5770 冁
5771 夔	5772 勹	5773 匍	5774 訇	5775 匐
5776 凫	5777 夙	5778 兕	5779 亠	5780 兖
5781 亳	5782 衮	5783 袤	5784 亵	5785 脔
5786 裒	5787 禀	5788 嬴	5789 蠃	5790 羸
5791 冫	5792 冱	5793 冽	5794 冼	

第 58 区

5801 凇	5802 冖	5803 冢	5804 冥	5805 讠
5806 讦	5807 讧	5808 讪	5809 讴	5810 讵
5811 讷	5812 诂	5813 诃	5814 诋	5815 诏
5816 诎	5817 诒	5818 诓	5819 诔	5820 诖

5821 诘 5822 诙 5823 诜 5824 诟 5825 诠
5826 诤 5827 诨 5828 诩 5829 诮 5830 诰
5831 诳 5832 诶 5833 诹 5834 诼 5835 诿
5836 谀 5837 谂 5838 谄 5839 谇 5840 谌
5841 谏 5842 谑 5843 谒 5844 谔 5845 谕
5846 谖 5847 谙 5848 谛 5849 谘 5850 谝
5851 谟 5852 谠 5853 谡 5854 谥 5855 谧
5856 谪 5857 谫 5858 谮 5859 谯 5860 谲
5861 谳 5862 谵 5863 谶 5864 卩 5865 卺
5866 阝 5867 阢 5868 阡 5869 阱 5870 阪
5871 阽 5872 阼 5873 陂 5874 陉 5875 陔
5876 陟 5877 陧 5878 陬 5879 陲 5880 陴
5881 隈 5882 隍 5883 隗 5884 隰 5885 邗
5886 邛 5887 邝 5888 邙 5889 邬 5890 邡
5891 邴 5892 邳 5893 邶 5894 邺

第 59 区

5901 邸 5902 邰 5903 郏 5904 郅 5905 邾
5906 郐 5907 郄 5908 郇 5909 郓 5910 郦
5911 郢 5912 郜 5913 郗 5914 郛 5915 郫
5916 郯 5917 郾 5918 鄄 5919 鄢 5920 鄞
5921 鄣 5922 鄱 5923 鄯 5924 鄹 5925 酃
5926 酆 5927 刍 5928 奂 5929 劢 5930 劬
5931 劭 5932 劾 5933 哿 5934 勐 5935 勖
5936 勰 5937 叟 5938 燮 5939 矍 5940 廴
5941 凵 5942 凼 5943 鬯 5944 厶 5945 弁
5946 畚 5947 巯 5948 坌 5949 垩 5950 垡

5951 塾　5952 墼　5953 壅　5954 壑　5955 圩
5956 圬　5957 圪　5958 圳　5959 圹　5960 圮
5961 圯　5962 坜　5963 圻　5964 坂　5965 坩
5966 垅　5967 坫　5968 垆　5969 坼　5970 坻
5971 坨　5972 坭　5973 坶　5974 坳　5975 垭
5976 垤　5977 垌　5978 垲　5979 埏　5980 垧
5981 垴　5982 垓　5983 垠　5984 埕　5985 埘
5986 埚　5987 埙　5988 埒　5989 垸　5990 埴
5991 埯　5992 埸　5993 埤　5994 埝

第 60 区

6001 堋　6002 堍　6003 埽　6004 埭　6005 堀
6006 堞　6007 堙　6008 塄　6009 堠　6010 塥
6011 塬　6012 墁　6013 墉　6014 墚　6015 墀
6016 馨　6017 鼙　6018 懿　6019 艹　6020 艽
6021 艿　6022 芏　6023 芊　6024 芨　6025 芄
6026 芎　6027 芑　6028 芗　6029 芙　6030 芫
6031 芸　6032 芾　6033 芰　6034 苈　6035 苊
6036 苣　6037 芘　6038 芷　6039 芮　6040 苋
6041 苌　6042 苁　6043 芩　6044 芴　6045 芡
6046 芪　6047 芟　6048 苄　6049 苎　6050 芤
6051 苡　6052 茉　6053 苷　6054 苤　6055 茏
6056 茇　6057 苜　6058 苴　6059 苒　6060 苘
6061 茌　6062 苻　6063 苓　6064 茑　6065 茚
6066 茆　6067 茔　6068 茕　6069 苠　6070 苕
6071 茜　6072 荑　6073 荛　6074 荜　6075 茈
6076 莒　6077 茼　6078 茴　6079 茱　6080 莛

6081 荞 6082 茯 6083 荏 6084 荇 6085 荃
6086 荟 6087 荀 6088 茗 6089 荠 6090 茭
6091 茺 6092 茳 6093 荦 6094 荥

第 61 区

6101 荨 6102 茛 6103 荩 6104 荬 6105 荪
6106 荭 6107 荮 6108 莰 6109 荸 6110 莳
6111 莴 6112 莠 6113 莪 6114 莓 6115 莜
6116 莅 6117 荼 6118 莶 6119 莩 6120 荽
6121 莸 6122 荻 6123 莘 6124 莞 6125 莨
6126 莺 6127 莼 6128 菁 6129 萁 6130 菥
6131 菘 6132 堇 6133 萘 6134 萋 6135 菝
6136 菽 6137 菖 6138 萜 6139 萸 6140 萑
6141 萆 6142 菔 6143 菟 6144 萏 6145 萃
6146 菸 6147 菹 6148 菪 6149 菅 6150 菀
6151 萦 6152 菰 6153 菡 6154 葜 6155 葑
6156 葚 6157 葙 6158 葳 6159 蒇 6160 蒈
6161 葺 6162 蒉 6163 葸 6164 萼 6165 葆
6166 葩 6167 葶 6168 蒌 6169 蒎 6170 萱
6171 葭 6172 蓁 6173 蓍 6174 蓐 6175 蓦
6176 蒽 6177 蓓 6178 蓊 6179 蒿 6180 蒺
6181 蓠 6182 蒡 6183 蒹 6184 蒴 6185 蒗
6186 蓥 6187 蓣 6188 蔌 6189 甍 6190 蔸
6191 蓰 6192 蔹 6193 蔟 6194 蔺

第 62 区

6201 蕖 6202 蔻 6203 蓿 6204 蓼 6205 蕙
6206 蕈 6207 蕨 6208 蕤 6209 蕞 6210 蕺
6211 瞢 6212 蕃 6213 蕲 6214 蕻 6215 薤
6216 薨 6217 薇 6218 薏 6219 蕹 6220 薮
6221 薜 6222 薅 6223 薹 6224 薷 6225 薰
6226 藓 6227 藁 6228 藜 6229 藿 6230 蘧
6231 蘅 6232 蘩 6233 蘖 6234 蘼 6235 廾
6236 弈 6237 夼 6238 奁 6239 耷 6240 奕
6241 奚 6242 奘 6243 匏 6244 尢 6245 尥
6246 尬 6247 尴 6248 扌 6249 扪 6250 抟
6251 抻 6252 拊 6253 拚 6254 拗 6255 拮
6256 挢 6257 拶 6258 挹 6259 捋 6260 捃
6261 掭 6262 揶 6263 捱 6264 捺 6265 掎
6266 掴 6267 捭 6268 掬 6269 掊 6270 捩
6271 掮 6272 掼 6273 揲 6274 揸 6275 揠
6276 揿 6277 揄 6278 揞 6279 揎 6280 摒
6281 揆 6282 掾 6283 摅 6284 摁 6285 搋
6286 搛 6287 搠 6288 搌 6289 搦 6290 搡
6291 摞 6292 撄 6293 摭 6294 撖

第 63 区

6301 摺 6302 撷 6303 撸 6304 撙 6305 撺
6306 擀 6307 擐 6308 擗 6309 擤 6310 擢
6311 攉 6312 攥 6313 攮 6314 弋 6315 忒
6316 甙 6317 弑 6318 卟 6319 叱 6320 叽

6321 叩	6322 叨	6323 叻	6324 吒	6325 吖
6326 吆	6327 呋	6328 呒	6329 呓	6330 呔
6331 呖	6332 呃	6333 吡	6334 呗	6335 呙
6336 吣	6337 吲	6338 咂	6339 咔	6340 呷
6341 呱	6342 呤	6343 咚	6344 咛	6345 咄
6346 呶	6347 呦	6348 咝	6349 哐	6350 咭
6351 哂	6352 咴	6353 哒	6354 咧	6355 咦
6356 哓	6357 哔	6358 呲	6359 咣	6360 哕
6361 咻	6362 咿	6363 哌	6364 哙	6365 哚
6366 哜	6367 咩	6368 咪	6369 咤	6370 哝
6371 哏	6372 哞	6373 唛	6374 哧	6375 唠
6376 哽	6377 唔	6378 哳	6379 唢	6380 唣
6381 唏	6382 唑	6383 唧	6384 唪	6385 啧
6386 喏	6387 喵	6388 啉	6389 啭	6390 啁
6391 啕	6392 唿	6393 啐	6394 唼	

第 64 区

6401 啮	6402 啖	6403 啵	6404 啶	6405 啷
6406 唳	6407 唰	6408 啜	6409 喋	6410 嗒
6411 喃	6412 喱	6413 喹	6414 喈	6415 喁
6416 喟	6417 啾	6418 嗖	6419 喑	6420 啻
6421 嗟	6422 喽	6423 喾	6424 喔	6425 喙
6426 嗪	6427 嗷	6428 嗉	6429 嘟	6430 嗑
6431 嗫	6432 嗬	6433 嗔	6434 嗦	6435 嗝
6436 嗄	6437 嗯	6438 嗥	6439 嗲	6440 嗳
6441 嗌	6442 嗍	6443 嗨	6444 嗵	6445 嗤
6446 辔	6447 嘞	6448 嘈	6449 嘌	6450 嘁

6451 嘤 6452 嘣 6453 嗾 6454 嘟 6455 嘧
6456 嘭 6457 噘 6458 嘹 6459 噗 6460 嘬
6461 噍 6462 噢 6463 噙 6464 噜 6465 噌
6466 噔 6467 嚆 6468 噤 6469 噱 6470 噫
6471 噻 6472 噼 6473 嚅 6474 嚓 6475 嚯
6476 囔 6477 囗 6478 囝 6479 囡 6480 囵
6481 囫 6482 囹 6483 囿 6484 圄 6485 圊
6486 圉 6487 圜 6488 帏 6489 帙 6490 帔
6491 帑 6492 帱 6493 帻 6494 帼

第 65 区

6501 帷 6502 幄 6503 幔 6504 幛 6505 幞
6506 幡 6507 岌 6508 屺 6509 岍 6510 岐
6511 岖 6512 岈 6513 岘 6514 岙 6515 岑
6516 岚 6517 岜 6518 岵 6519 岢 6520 岽
6521 岬 6522 岫 6523 岱 6524 岣 6525 峁
6526 岷 6527 峄 6528 峒 6529 峤 6530 峋
6531 峥 6532 崂 6533 崃 6534 崧 6535 崦
6536 崮 6537 崤 6538 崞 6539 崆 6540 崛
6541 嵘 6542 崾 6543 崴 6544 崽 6545 嵬
6546 嵛 6547 嵯 6548 嵝 6549 嵫 6550 嵋
6551 嵊 6552 嵩 6553 嵴 6554 嶂 6555 嶙
6556 嶝 6557 豳 6558 嶷 6559 巅 6560 彳
6561 彷 6562 徂 6563 徇 6564 徉 6565 後
6566 徕 6567 徙 6568 徜 6569 徨 6570 徭
6571 徵 6572 徼 6573 衢 6574 彡 6575 犭
6576 犰 6577 犴 6578 犷 6579 犸 6580 狃

6581 狁 6582 狎 6583 狍 6584 狒 6585 狨
6586 狯 6587 狩 6588 狲 6589 狴 6590 狷
6591 猁 6592 狳 6593 猃 6594 狺

第 66 区

6601 狻 6602 猗 6603 猓 6604 猡 6605 猊
6606 猞 6607 猝 6608 猕 6609 猢 6610 猹
6611 猥 6612 猬 6613 猸 6614 猱 6615 獐
6616 獍 6617 獗 6618 獠 6619 獬 6620 獯
6621 獾 6622 舛 6623 夥 6624 飧 6625 夤
6626 夂 6627 饣 6628 饧 6629 饨 6630 饩
6631 饪 6632 饫 6633 饬 6634 饴 6635 饷
6636 饽 6637 馀 6638 馄 6639 馇 6640 馊
6641 馍 6642 馐 6643 馑 6644 馓 6645 馔
6646 馕 6647 庀 6648 庑 6649 庋 6650 庖
6651 庥 6652 庠 6653 庹 6654 庵 6655 庾
6656 庳 6657 赓 6658 廒 6659 廑 6660 廛
6661 廨 6662 廪 6663 膺 6664 忄 6665 忉
6666 忖 6667 忏 6668 怃 6669 忮 6670 怄
6671 忡 6672 忤 6673 忾 6674 怅 6675 怆
6676 忪 6677 忭 6678 忸 6679 怙 6680 怵
6681 怦 6682 怛 6683 怏 6684 怍 6685 怩
6686 怫 6687 怊 6688 怿 6689 怡 6690 恸
6691 恹 6692 恻 6693 恺 6694 恂

第 67 区

6701 恪 6702 恽 6703 悖 6704 悚 6705 悭
6706 悝 6707 悃 6708 悒 6709 悌 6710 悛
6711 惬 6712 悻 6713 悱 6714 惝 6715 惘
6716 惆 6717 惚 6718 悴 6719 愠 6720 愦
6721 愕 6722 愣 6723 惴 6724 愀 6725 愎
6726 愫 6727 慊 6728 慵 6729 憬 6730 憔
6731 憧 6732 憷 6733 懔 6734 懵 6735 忝
6736 隳 6737 闩 6738 闫 6739 闱 6740 闳
6741 闵 6742 闶 6743 闼 6744 闾 6745 阃
6746 阄 6747 阆 6748 阈 6749 阊 6750 阋
6751 阌 6752 阍 6753 阏 6754 阒 6755 阕
6756 阖 6757 阗 6758 阙 6759 阚 6760 丬
6761 爿 6762 戕 6763 氵 6764 汔 6765 汜
6766 汊 6767 沣 6768 沅 6769 沐 6770 沔
6771 沌 6772 汨 6773 汩 6774 汴 6775 汶
6776 沆 6777 沩 6778 泐 6779 泔 6780 沭
6781 泷 6782 泸 6783 泱 6784 泗 6785 沲
6786 泠 6787 泖 6788 泺 6789 泫 6790 泮
6791 沱 6792 泓 6793 泯 6794 泾

第 68 区

6801 洹 6802 洧 6803 洌 6804 浃 6805 浈
6806 洇 6807 洄 6808 洙 6809 洎 6810 洫
6811 浍 6812 洮 6813 洵 6814 洚 6815 浏
6816 浒 6817 浔 6818 洳 6819 涑 6820 浯

6821 涞 6822 涠 6823 浞 6824 涓 6825 涔
6826 浜 6827 浠 6828 浼 6829 浣 6830 渚
6831 淇 6832 淅 6833 淞 6834 渎 6835 涿
6836 淠 6837 渑 6838 淦 6839 淝 6840 淙
6841 渖 6842 涫 6843 渌 6844 涮 6845 渫
6846 湮 6847 湎 6848 湫 6849 溲 6850 湟
6851 溆 6852 湓 6853 湔 6854 渲 6855 渥
6856 湄 6857 滟 6858 溱 6859 溘 6860 滠
6861 漭 6862 滢 6863 溥 6864 溧 6865 溽
6866 溻 6867 溷 6868 滗 6869 溴 6870 滏
6871 溏 6872 滂 6873 溟 6874 潢 6875 潆
6876 潇 6877 漤 6878 漕 6879 滹 6880 漯
6881 漶 6882 潋 6883 潴 6884 漪 6885 漉
6886 漩 6887 澉 6888 澍 6889 澌 6890 潸
6891 潲 6892 潼 6893 潺 6894 濑

第 69 区

6901 濉 6902 澧 6903 澹 6904 澶 6905 濂
6906 濡 6907 濮 6908 濞 6909 濠 6910 濯
6911 瀚 6912 瀣 6913 瀛 6914 瀹 6915 瀵
6916 灏 6917 灞 6918 宀 6919 宄 6920 宕
6921 宓 6922 宥 6923 宸 6924 甯 6925 骞
6926 搴 6927 寤 6928 寮 6929 褰 6930 寰
6931 蹇 6932 謇 6933 辶 6934 迓 6935 迕
6936 迥 6937 迮 6938 迤 6939 迩 6940 迦
6941 迳 6942 迨 6943 逅 6944 逄 6945 逋

6946 邐　6947 逑　6948 逍　6949 逖　6950 逡
6951 逵　6952 逶　6953 逭　6954 逯　6955 遄
6956 遑　6957 遒　6958 遐　6959 遨　6960 遘
6961 遢　6962 遛　6963 暹　6964 遴　6965 遽
6966 邂　6967 邈　6968 邃　6969 邋　6970 彐
6971 彗　6972 彖　6973 彘　6974 尻　6975 咫
6976 屐　6977 屙　6978 孱　6979 屣　6980 屨
6981 羼　6982 弳　6983 弩　6984 弭　6985 艴
6986 弼　6987 鬻　6988 屮　6989 妁　6990 妃
6991 妍　6992 妩　6993 嫗　6994 妣

第 70 区

7001 妗　7002 姊　7003 妫　7004 妞　7005 妤
7006 姒　7007 妲　7008 妯　7009 姗　7010 妾
7011 婭　7012 嬈　7013 姝　7014 娈　7015 姣
7016 姘　7017 姹　7018 娌　7019 娉　7020 媧
7021 嫻　7022 娑　7023 娣　7024 娓　7025 婀
7026 婧　7027 婊　7028 婕　7029 娼　7030 婢
7031 嬋　7032 胬　7033 媼　7034 媛　7035 婷
7036 婺　7037 媾　7038 嫫　7039 媲　7040 嬡
7041 嬪　7042 媸　7043 嫠　7044 嫣　7045 嬙
7046 嫖　7047 嫦　7048 嫘　7049 嫜　7050 嬉
7051 嬗　7052 嬖　7053 嬲　7054 嬤　7055 孀
7056 尕　7057 尜　7058 孚　7059 孥　7060 孳
7061 孑　7062 孓　7063 孢　7064 駔　7065 駟
7066 駙　7067 騶　7068 驛　7069 駑　7070 駘
7071 驍　7072 驊　7073 駢　7074 驪　7075 騏
7076 騍　7077 騅　7078 驂　7079 騭　7080 騖

7081 骛 7082 骝 7083 骟 7084 骠 7085 骢
7086 骣 7087 骥 7088 骧 7089 纟 7090 纡
7091 纣 7092 纥 7093 纨 7094 纩

第 71 区

7101 纭 7102 纰 7103 纾 7104 绀 7105 绁
7106 绂 7107 绉 7108 绋 7109 绌 7110 绐
7111 绔 7112 绗 7113 绛 7114 绠 7115 绡
7116 绨 7117 绫 7118 绮 7119 绯 7120 绱
7121 绲 7122 缍 7123 绶 7124 绺 7125 绻
7126 绾 7127 缁 7128 缂 7129 缃 7130 缇
7131 缈 7132 缋 7133 缌 7134 缏 7135 缑
7136 缒 7137 缗 7138 缙 7139 缜 7140 缛
7141 缟 7142 缡 7143 缢 7144 缣 7145 缤
7146 缥 7147 缦 7148 缧 7149 缪 7150 缫
7151 缬 7152 缭 7153 缯 7154 缰 7155 缱
7156 缲 7157 缳 7158 缵 7159 幺 7160 畿
7161 巛 7162 甾 7163 邕 7164 玎 7165 玑
7166 玮 7167 玢 7168 玟 7169 珏 7170 珂
7171 珑 7172 玷 7173 玳 7174 珀 7175 珉
7176 珈 7177 珥 7178 珙 7179 顼 7180 琊
7181 珩 7182 珧 7183 珞 7184 玺 7185 珲
7186 琏 7187 琪 7188 瑛 7189 琦 7190 琥
7191 琨 7192 琰 7193 琮 7194 琬

第 72 区

7201 琛 7202 琚 7203 瑁 7204 瑜 7205 瑗
7206 瑕 7207 瑙 7208 瑷 7209 瑭 7210 瑾

7211 璜 7212 璎 7213 璀 7214 璁 7215 璇
7216 璋 7217 璞 7218 璨 7219 璩 7220 璐
7221 璧 7222 瓒 7223 璺 7224 韪 7225 韫
7226 韬 7227 杌 7228 杓 7229 杞 7230 杈
7231 杩 7232 枥 7233 枇 7234 杪 7235 杳
7236 枘 7237 枧 7238 杵 7239 枨 7240 枞
7241 枭 7242 枋 7243 杷 7244 杼 7245 柰
7246 栉 7247 柘 7248 栊 7249 柩 7250 枰
7251 栌 7252 柙 7253 枵 7254 柚 7255 枳
7256 柝 7257 栀 7258 柃 7259 枸 7260 柢
7261 栎 7262 柁 7263 柽 7264 栲 7265 栳
7266 桠 7267 桡 7268 桎 7269 桢 7270 桄
7271 桤 7272 梃 7273 栝 7274 桕 7275 桦
7276 桁 7277 桧 7278 桀 7279 栾 7280 桊
7281 桉 7282 栩 7283 梵 7284 梏 7285 桴
7286 桷 7287 梓 7288 桫 7289 棂 7290 楮
7291 棼 7292 椟 7293 椠 7294 棹

第 73 区

7301 椤 7302 棰 7303 椋 7304 椁 7305 楗
7306 棣 7307 椐 7308 楱 7309 椹 7310 楠
7311 楂 7312 楝 7313 榄 7314 楫 7315 榀
7316 榘 7317 楸 7318 椴 7319 槌 7320 榇
7321 榈 7322 槎 7323 榉 7324 楦 7325 楣
7326 楹 7327 榛 7328 榧 7329 榻 7330 榫
7331 榭 7332 槔 7333 榱 7334 槁 7335 槊
7336 槟 7337 榕 7338 槠 7339 榍 7340 槿
7341 樯 7342 槭 7343 樗 7344 樘 7345 橥

7346 槲 7347 橄 7348 樾 7349 檠 7350 橐
7351 橛 7352 樵 7353 檎 7354 橹 7355 樽
7356 樨 7357 橘 7358 橼 7359 檑 7360 檐
7361 檩 7362 檗 7363 檫 7364 猷 7365 獒
7366 殁 7367 殂 7368 殇 7369 殄 7370 殒
7371 殓 7372 殍 7373 殚 7374 殛 7375 殡
7376 殪 7377 轫 7378 轭 7379 轱 7380 轲
7381 轳 7382 轵 7383 轶 7384 轸 7385 轷
7386 轹 7387 轺 7388 轼 7389 轾 7390 辁
7391 辂 7392 辄 7393 辇 7394 辋

第 74 区

7401 辍 7402 辎 7403 辏 7404 辘 7405 辚
7406 軎 7407 戋 7408 戗 7409 戛 7410 戟
7411 戢 7412 戡 7413 戥 7414 戤 7415 戬
7416 臧 7417 瓯 7418 瓴 7419 瓿 7420 甏
7421 甑 7422 甓 7423 攴 7424 旮 7425 旯
7426 旰 7427 昊 7428 昙 7429 杲 7430 昃
7431 昕 7432 昀 7433 炅 7434 曷 7435 昝
7436 昴 7437 昱 7438 昶 7439 昵 7440 耆
7441 晟 7442 晔 7443 晁 7444 晏 7445 晖
7446 晡 7447 晗 7448 晷 7449 暄 7450 暌
7451 暧 7452 暝 7453 暾 7454 曛 7455 曜
7456 曦 7457 曩 7458 贲 7459 贳 7460 贶
7461 贻 7462 贽 7463 赀 7464 赅 7465 赆
7466 赈 7467 赉 7468 赇 7469 赍 7470 赕
7471 赙 7472 觇 7473 觊 7474 觋 7475 觌

7476 觎 7477 觏 7478 觐 7479 觑 7480 牮
7481 犟 7482 牝 7483 牦 7484 牯 7485 牾
7486 牿 7487 犄 7488 犋 7489 犍 7490 犏
7491 犒 7492 挈 7493 挲 7494 掰

第 75 区

7501 搿 7502 擘 7503 耄 7504 毪 7505 毳
7506 毽 7507 毵 7508 毹 7509 氅 7510 氇
7511 氆 7512 氍 7513 氕 7514 氘 7515 氙
7516 氚 7517 氡 7518 氩 7519 氤 7520 氪
7521 氲 7522 攵 7523 敕 7524 敫 7525 牍
7526 牒 7527 牖 7528 爰 7529 虢 7530 刖
7531 肟 7532 肜 7533 肓 7534 肼 7535 朊
7536 肽 7537 肱 7538 肫 7539 肭 7540 肴
7541 肷 7542 胧 7543 胨 7544 胩 7545 胪
7546 胛 7547 胂 7548 胄 7549 胙 7550 胍
7551 胗 7552 朐 7553 胝 7554 胫 7555 胱
7556 胴 7557 胭 7558 脍 7559 脎 7560 胲
7561 胼 7562 朕 7563 脒 7564 豚 7565 脶
7566 脞 7567 脬 7568 脘 7569 脲 7570 腈
7571 腌 7572 腓 7573 腴 7574 腙 7575 腚
7576 腱 7577 腠 7578 腩 7579 腼 7580 腽
7581 腭 7582 腧 7583 塍 7584 媵 7585 膈
7586 膂 7587 膑 7588 滕 7589 膣 7590 膪
7591 臌 7592 朦 7593 臊 7594 膻

第 76 区

7601 臁 7602 膦 7603 欤 7604 欷 7605 欹

7606 歃　7607 歆　7608 歙　7609 飑　7610 飒
7611 飓　7612 飕　7613 飙　7614 飚　7615 殳
7616 彀　7617 毂　7618 觳　7619 斐　7620 齑
7621 斓　7622 於　7623 旆　7624 旄　7625 旃
7626 旌　7627 旎　7628 旒　7629 旖　7630 炀
7631 炜　7632 炖　7633 炝　7634 炻　7635 烀
7636 炷　7637 炫　7638 炱　7639 烨　7640 烊
7641 焐　7642 焓　7643 焖　7644 焯　7645 焱
7646 煳　7647 煜　7648 煨　7649 煅　7650 煲
7651 煊　7652 煸　7653 煺　7654 熘　7655 熳
7656 熵　7657 熨　7658 熠　7659 燠　7660 燔
7661 燧　7662 燹　7663 爝　7664 爨　7665 灬
7666 焘　7667 煦　7668 熹　7669 戾　7670 戽
7671 扃　7672 扈　7673 扉　7674 礻　7675 祀
7676 祆　7677 祉　7678 祛　7679 祜　7680 祓
7681 祚　7682 祢　7683 祗　7684 祠　7685 祯
7686 祧　7687 祺　7688 禅　7689 禊　7690 禚
7691 禧　7692 禳　7693 忑　7694 忐

第 77 区

7701 怼　7702 恝　7703 恚　7704 恧　7705 恁
7706 恙　7707 恣　7708 悫　7709 愆　7710 愍
7711 慝　7712 憩　7713 憝　7714 懋　7715 懑
7716 戆　7717 肀　7718 聿　7719 沓　7720 泶
7721 淼　7722 矶　7723 矸　7724 砀　7725 砉
7726 砗　7727 砘　7728 砑　7729 斫　7730 砭
7731 砜　7732 砝　7733 砹　7734 砺　7735 砻

7736 砟 7737 砼 7738 砥 7739 砬 7740 砣
7741 砩 7742 硎 7743 硭 7744 硖 7745 硗
7746 砦 7747 硐 7748 硇 7749 硌 7750 硪
7751 碛 7752 碓 7753 碚 7754 碇 7755 碜
7756 碡 7757 碣 7758 碲 7759 碹 7760 碥
7761 磔 7762 磙 7763 磉 7764 磬 7765 磲
7766 礅 7767 磴 7768 礓 7769 礤 7770 礞
7771 礴 7772 龛 7773 黹 7774 黻 7775 黼
7776 盱 7777 眄 7778 眍 7779 盹 7780 眇
7781 眈 7782 眚 7783 眢 7784 眙 7785 眭
7786 眦 7787 眵 7788 眸 7789 睐 7790 睑
7791 睇 7792 睃 7793 睚 7794 睨

第 78 区

7801 睢 7802 睥 7803 睿 7804 瞍 7805 睽
7806 瞀 7807 瞌 7808 瞑 7809 瞟 7810 瞠
7811 瞰 7812 瞵 7813 瞽 7814 町 7815 畀
7816 畎 7817 畋 7818 畈 7819 畛 7820 畲
7821 畹 7822 疃 7823 罘 7824 罡 7825 罟
7826 詈 7827 罨 7828 罴 7829 罱 7830 罹
7831 羁 7832 罾 7833 盍 7834 盥 7835 蠲
7836 钅 7837 钆 7838 钇 7839 钋 7840 钊
7841 钌 7842 钍 7843 钏 7844 钐 7845 钔
7846 钗 7847 钕 7848 钚 7849 钛 7850 钜
7851 钣 7852 钤 7853 钫 7854 钪 7855 钭
7856 钬 7857 钯 7858 钰 7859 钲 7860 钴
7861 钶 7862 钷 7863 钸 7864 钹 7865 钺
7866 钼 7867 钽 7868 钿 7869 铄 7870 铈

7871 铉 7872 铊 7873 铋 7874 铌 7875 铍
7876 铎 7877 铐 7878 铑 7879 铒 7880 铕
7881 铖 7882 铗 7883 铙 7884 铘 7885 铛
7886 铞 7887 铟 7888 铠 7889 铢 7890 铤
7891 铥 7892 铧 7893 铨 7894 铪

第 79 区

7901 铩 7902 铫 7903 铮 7904 铯 7905 铳
7906 铴 7907 铵 7908 铷 7909 铹 7910 铼
7911 铽 7912 铿 7913 锃 7914 锂 7915 锆
7916 锇 7917 锉 7918 锊 7919 锍 7920 锎
7921 锏 7922 锒 7923 锓 7924 锔 7925 锕
7926 锖 7927 锘 7928 锛 7929 锝 7930 锞
7931 锟 7932 锢 7933 锪 7934 锫 7935 锩
7936 锬 7937 锱 7938 锲 7939 锴 7940 锶
7941 锷 7942 锸 7943 锼 7944 锾 7945 锿
7946 镂 7947 锵 7948 镄 7949 镅 7950 镆
7951 镉 7952 镌 7953 镎 7954 镏 7955 镒
7956 镓 7957 镔 7958 镖 7959 镗 7960 镘
7961 镙 7962 镛 7963 镞 7964 镟 7965 镝
7966 镡 7967 镢 7968 镤 7969 镥 7970 镦
7971 镧 7972 镨 7973 镩 7974 镪 7975 镫
7976 镬 7977 镯 7978 镱 7979 镲 7980 镳
7981 锺 7982 矧 7983 矬 7984 雉 7985 秕
7986 秭 7987 秣 7988 秫 7989 稆 7990 嵇
7991 稃 7992 稂 7993 稞 7994 稔

第 80 区

8001 稹	8002 稷	8003 穑	8004 黏	8005 馥
8006 穰	8007 皈	8008 皎	8009 皓	8010 皙
8011 皤	8012 瓞	8013 瓠	8014 甬	8015 鸠
8016 鸢	8017 鸨	8018 鸩	8019 鸪	8020 鸫
8021 鸬	8022 鸲	8023 鸱	8024 鸶	8025 鸸
8026 鸷	8027 鸹	8028 鸺	8029 鸾	8030 鹁
8031 鹂	8032 鹄	8033 鹆	8034 鹇	8035 鹈
8036 鹉	8037 鹋	8038 鹌	8039 鹎	8040 鹑
8041 鹕	8042 鹗	8043 鹚	8044 鹛	8045 鹜
8046 鹞	8047 鹣	8048 鹦	8049 鹧	8050 鹨
8051 鹩	8052 鹪	8053 鹫	8054 鹬	8055 鹱
8056 鹭	8057 鹳	8058 疒	8059 疔	8060 疖
8061 疠	8062 疝	8063 疬	8064 疣	8065 疳
8066 疴	8067 疸	8068 痄	8069 疱	8070 疰
8071 痃	8072 痂	8073 痖	8074 痍	8075 痣
8076 痨	8077 痦	8078 痤	8079 痫	8080 痧
8081 瘃	8082 痱	8083 痼	8084 痿	8085 瘐
8086 瘀	8087 瘅	8088 瘌	8089 瘗	8090 瘊
8091 瘥	8092 瘘	8093 瘕	8094 瘙	

第 81 区

8101 瘛	8102 瘼	8103 瘢	8104 瘠	8105 癀
8106 瘭	8107 瘰	8108 瘿	8109 瘵	8110 癃
8111 瘾	8112 瘳	8113 癍	8114 癞	8115 癔
8116 癜	8117 癖	8118 癫	8119 癯	8120 翊
8121 竦	8122 穸	8123 穹	8124 窀	8125 窆

8126 窈 8127 窕 8128 窦 8129 窠 8130 窬
8131 窨 8132 窭 8133 窳 8134 衤 8135 衩
8136 衲 8137 衽 8138 衿 8139 袂 8140 袢
8141 裆 8142 袷 8143 袼 8144 裉 8145 裢
8146 裎 8147 裣 8148 裥 8149 裱 8150 褚
8151 裼 8152 裨 8153 裾 8154 裰 8155 褡
8156 褙 8157 褓 8158 褛 8159 褊 8160 褴
8161 褫 8162 褶 8163 襁 8164 襦 8165 襻
8166 疋 8167 胥 8168 皲 8169 皴 8170 矜
8171 耒 8172 耔 8173 耖 8174 耜 8175 耠
8176 耢 8177 耥 8178 耦 8179 耧 8180 耩
8181 耨 8182 耱 8183 耋 8184 耵 8185 聃
8186 聆 8187 聍 8188 聒 8189 聩 8190 聱
8191 覃 8192 顸 8193 颀 8194 颃

第 82 区

8201 颉 8202 颌 8203 颍 8204 颏 8205 颔
8206 颚 8207 颛 8208 颞 8209 颟 8210 颡
8211 颢 8212 颥 8213 颦 8214 虍 8215 虔
8216 虬 8217 虮 8218 虿 8219 虺 8220 虼
8221 虻 8222 蚨 8223 蚍 8224 蚋 8225 蚬
8226 蚝 8227 蚧 8228 蚣 8229 蚪 8230 蚓
8231 蚩 8232 蚶 8233 蛄 8234 蚵 8235 蛎
8236 蚰 8237 蚺 8238 蚱 8239 蚯 8240 蛉
8241 蛏 8242 蚴 8243 蛩 8244 蛱 8245 蛲
8246 蛭 8247 蛳 8248 蛐 8249 蜓 8250 蛞
8251 蛴 8252 蛟 8253 蛘 8254 蛑 8255 蜃

8256 蜇 8257 蛸 8258 蜈 8259 蜊 8260 蜍
8261 蜉 8262 蜣 8263 蜻 8264 蜞 8265 蜥
8266 蜮 8267 蜚 8268 蜾 8269 蝈 8270 蜴
8271 蜱 8272 蜩 8273 蜷 8274 蜿 8275 螂
8276 蜢 8277 蝽 8278 蝾 8279 蝻 8280 蝠
8281 蝰 8282 蝌 8283 蝮 8284 螋 8285 蝓
8286 蝣 8287 蝼 8288 蝤 8289 蝙 8290 蝥
8291 螓 8292 螯 8293 螨 8294 蟒

第 83 区

8301 蟆 8302 螈 8303 螅 8304 螭 8305 螗
8306 螃 8307 螫 8308 蟥 8309 螬 8310 螵
8311 螳 8312 蟋 8313 蟓 8314 螽 8315 蟑
8316 蟀 8317 蟊 8318 蟛 8319 蟪 8320 蟠
8321 蟮 8322 蠖 8323 蠓 8324 蟾 8325 蠊
8326 蠛 8327 蠡 8328 蠹 8329 蠼 8330 缶
8331 罂 8332 罄 8333 罅 8334 舐 8335 竺
8336 竽 8337 笈 8338 笃 8339 笄 8340 笕
8341 笊 8342 笫 8343 笏 8344 筇 8345 笸
8346 笪 8347 笙 8348 笮 8349 笱 8350 笠
8351 笥 8352 笤 8353 笳 8354 笾 8355 笞
8356 筘 8357 筚 8358 筅 8359 筵 8360 筌
8361 筝 8362 筠 8363 筮 8364 筻 8365 筢
8366 筲 8367 筱 8368 箐 8369 箦 8370 箧
8371 箸 8372 箬 8373 箝 8374 箨 8375 箅
8376 箪 8377 箜 8378 箢 8379 箫 8380 箴
8381 篑 8382 篁 8383 篌 8384 篝 8385 篚

8386 篥 8387 篦 8388 篪 8389 簌 8390 篾
8391 篼 8392 簏 8393 簖 8394 簋

第 84 区

8401 簟 8402 簪 8403 簦 8404 簸 8405 籁
8406 籀 8407 臾 8408 舁 8409 舂 8410 舄
8411 臬 8412 衄 8413 舡 8414 舢 8415 舣
8416 舭 8417 舯 8418 舨 8419 舫 8420 舸
8421 舻 8422 舳 8423 舴 8424 舾 8425 艄
8426 艉 8427 艋 8428 艏 8429 艚 8430 艟
8431 艨 8432 衾 8433 袅 8434 袈 8435 裘
8436 裟 8437 襞 8438 羝 8439 羟 8440 羧
8441 羯 8442 羰 8443 羲 8444 籼 8445 敉
8446 粑 8447 粝 8448 粜 8449 粞 8450 粢
8451 粲 8452 粼 8453 粽 8454 糁 8455 糇
8456 糌 8457 糍 8458 糈 8459 糅 8460 糗
8461 糨 8462 艮 8463 暨 8464 羿 8465 翎
8466 翕 8467 翥 8468 翡 8469 翦 8470 翩
8471 翮 8472 翳 8473 糸 8474 絷 8475 綦
8476 綮 8477 繇 8478 纛 8479 麸 8480 麴
8481 赳 8482 趄 8483 趔 8484 趑 8485 趱
8486 赧 8487 赭 8488 豇 8489 豉 8490 酊
8491 酐 8492 酎 8493 酏 8494 酤

第 85 区

8501 酢 8502 酡 8503 酰 8504 酩 8505 酯
8506 酽 8507 酾 8508 酲 8509 酴 8510 酹
8511 醌 8512 醅 8513 醐 8514 醍 8515 醑

8516 醢	8517 醣	8518 醪	8519 醭	8520 醮
8521 醯	8522 醵	8523 醴	8524 醺	8525 豕
8526 鹾	8527 趸	8528 跫	8529 踅	8530 蹙
8531 蹩	8532 趵	8533 趿	8534 趼	8535 趺
8536 跄	8537 跖	8538 跗	8539 跚	8540 跞
8541 跎	8542 跏	8543 跛	8544 跆	8545 跬
8546 跷	8547 跸	8548 跣	8549 跹	8550 跻
8551 跤	8552 踉	8553 跽	8554 踔	8555 踝
8556 踟	8557 踬	8558 踮	8559 踣	8560 踯
8561 踺	8562 蹀	8563 踹	8564 踵	8565 踽
8566 踱	8567 蹉	8568 蹁	8569 蹂	8570 蹑
8571 蹒	8572 蹊	8573 蹰	8574 蹶	8575 蹼
8576 蹯	8577 蹴	8578 躅	8579 躏	8580 躔
8581 躐	8582 躜	8583 躞	8584 豸	8585 貂
8586 貊	8587 貅	8588 貘	8589 貔	8590 斛
8591 觖	8592 觞	8593 觚	8594 觜	

第 86 区

8601 觥	8602 觫	8603 觯	8604 訾	8605 謦
8606 靓	8607 雩	8608 雳	8609 雯	8610 霆
8611 霁	8612 霈	8613 霏	8614 霎	8615 霪
8616 霭	8617 霰	8618 霾	8619 龀	8620 龃
8621 龅	8622 龆	8623 龇	8624 龈	8625 龉
8626 龊	8627 龌	8628 黾	8629 鼋	8630 鼍
8631 隹	8632 隼	8633 隽	8634 雎	8635 雒
8636 瞿	8637 雠	8638 銎	8639 銮	8640 鋈
8641 錾	8642 鍪	8643 鏊	8644 鎏	8645 鐾

8646 鑫 8647 鱿 8648 鲂 8649 鲅 8650 鲆
8651 鲇 8652 鲈 8653 稣 8654 鲋 8655 鲎
8656 鲐 8657 鲑 8658 鲒 8659 鲔 8660 鲕
8661 鲚 8662 鲛 8663 鲞 8664 鲟 8665 鲠
8666 鲡 8667 鲢 8668 鲣 8669 鲥 8670 鲦
8671 鲧 8672 鲨 8673 鲩 8674 鲫 8675 鲭
8676 鲮 8677 鲰 8678 鲱 8679 鲲 8680 鲳
8681 鲴 8682 鲵 8683 鲶 8684 鲷 8685 鲺
8686 鲻 8687 鲼 8688 鲽 8689 鳄 8690 鳅
8691 鳆 8692 鳇 8693 鳊 8694 鳋

第 87 区

8701 鳌 8702 鳍 8703 鳎 8704 鳏 8705 鳐
8706 鳓 8707 鳔 8708 鳕 8709 鳗 8710 鳘
8711 鳙 8712 鳜 8713 鳝 8714 鳟 8715 鳢
8716 靼 8717 鞅 8718 鞑 8719 鞒 8720 鞔
8721 鞯 8722 鞫 8723 鞣 8724 鞲 8725 鞴
8726 骱 8727 骰 8728 骷 8729 鹘 8730 骶
8731 骺 8732 骼 8733 髁 8734 髀 8735 髅
8736 髂 8737 髋 8738 髌 8739 髑 8740 魅
8741 魃 8742 魇 8743 魉 8744 魈 8745 魍
8746 魑 8747 飨 8748 餍 8749 餮 8750 饕
8751 饔 8752 髟 8753 髡 8754 髦 8755 髯
8756 髫 8757 髻 8758 髭 8759 髹 8760 鬈
8761 鬏 8762 鬓 8763 鬟 8764 鬣 8765 麽
8766 麾 8767 縻 8768 麂 8769 麇 8770 麈
8771 麋 8772 麒 8773 鏖 8774 麝 8775 麟

8776 黛 8777 黜 8778 黝 8779 黠 8780 黟

8781 黢 8782 黩 8783 黧 8784 黥 8785 黪

8786 黯 8787 黔 8788 [illegible] 8789 [illegible] 8790 [illegible]

8791 [illegible] 8792 黹 8793 黻 8794 黼

科技新书目：213-204　定价：7.40元

ISBN 7-5029-0329-1/TP·0017


中华学习机汉字软件
气象出版社